Bayesian parameter estimation of a $k - \epsilon$ model for accurate jet-in-crossflow simulations

Jaideep Ray[a] and Sophia Lefantzi[b]
Sandia National Laboratories, Livermore, CA 94550-0969,USA

Srinivasan Arunajatesan[c] and Lawrence Dechant[d]
Sandia National Laboratories, Albuquerque, NM 87185-5800, USA

Reynolds-Averaged Navier-Stokes (RANS) models are not very accurate for high Reynolds number, compressible jet-in-crossflow interactions. The inaccuracy arises from the use of inappropriate model parameters and model-form error in the RANS model. In this work we pursue the hypothesis that RANS predictions could be significantly improved by using parameters inferred from experimental measurements of a supersonic jet interacting with a transonic crossflow. We formulate a Bayesian inverse problem to estimate 3 RANS parameters $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ and use a Markov chain Monte Carlo (MCMC) method to develop a probability density function for them. The cost of MCMC is addressed by developing statistical surrogates for the RANS models. We find that only a subset of the $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ space, $\mathcal{R}$, supports realistic flow simulations. We use $\mathcal{R}$ as our prior belief when formulating the inverse problem. It is enforced with a classifier in our MCMC solution. We find that the calibrated parameters improve predictions of the entire flowfield substantially, compared to the nominal/literature values of $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$; further, this improvement is seen to hold for interactions at other Mach numbers and jet strengths for which we have experimental data to provide a comparison. We also quantify the residual error, which is an approximation of the model-form error; it is most easily measured in terms of turbulent stresses.

---

[a] Technical Staff, Quantitative Modeling and Analysis, MS 9159
[b] Technical Staff, Quantitative Modeling and Analysis, MS 9152, Senior Member
[c] Technical Staff, Aerosciences Department, MS 0825, Senior Member
[d] Technical Staff, Aerosciences Department, MS 0825, Senior Member

Nomenclature

$\mathbf{C}$ = Parameters in the $k-\epsilon$ RANS model to be calibrated

$\mathbf{C}_{nom}$ = Nominal values of $\mathbf{C}$

C_μ = A parameter in the eddy-viscosity sub-model in a $k-\epsilon$ RANS model

$C_{\epsilon 2}, C_{\epsilon 1}$ = Parameters in the equation of the evolution of ϵ in a $k-\epsilon$ RANS model

δ_m = Structural error; difference between experimental data and calibrated surrogate model predictions

$\mathcal{N}(\mu, \sigma^2)$ = Normal distribution with mean μ and standard deviation σ

$\mathcal{P}_f$ = Set of probes where experimental vorticity measurements are available

$\mathcal{P}_*$ = Subset of $\mathcal{P}_f$ with large vorticity where numerical noise is insignificant

$\mathcal{P}$ = Subset of $\mathcal{P}_*$ where accurate surrogate models can be made

$\mathcal{R}$ = The physically relevant part of the $\mathbf{C}$ parameter space

d = Approximation error in the surrogate model

J = jet-to-crossflow momentum ratio

$y^{(p)}$ = RANS prediction at probe p

$y_s^{(p)}$ = Surrogate model prediction at probe p

AIC = Akaike Information Criterion

GP = Gaussian Process

JIC = Jet-in-Crossflow

JPDF = Joint probability density function

LS = Learning Set

MAP = maximum *a posteriori*; location of max. value of posterior probability density

MCMC = Markov chain Monte Carlo

RANS = Reynolds-Averaged Navier-Stokes

SVMC = Support vector machine classifier

TS = Testing Set

I. Introduction

Reynolds-Averaged Navier-Stokes (RANS) models [1] are routinely used in aerodynamics due to their robustness and speed vis-à-vis other simulation methods e.g., Large Eddy Simulations. Typically, they are used in conjunction with $k-\epsilon$ equations to model the evolution of the turbulent kinetic energy and its dissipation rate. The equations for k and ϵ contain empirical "constitutive relationships", e.g., the eddy viscosity model mapping strain-rates to turbulent stresses, the model for turbulent viscosity in terms of k and ϵ and empirical constants (parameters). The values of these parameters are deemed to be universal and obtained by calibrating to canonical flows e.g., channel and shear flows [2–4]. We call these the "nominal" values of the parameters. However, there is nothing sacrosanct about these parameters' values and there have been attempts to tune them [2, 5–9] for specific flows. The tuned parameters can vary quite substantially from the nominal ones.

Despite their widespread use, the predictive skill of $k-\epsilon$ simulations leave much to be desired. This is especially true for complex turbulent flows, such as jet-in-crossflow interactions at high Reynolds numbers. The lack of predictive accuracy has two sources: (1) the structural or model-form error, due to approximations inherent in the RANS $k-\epsilon$ formulation i.e., errors due to missing physics and (2) the use of sub-optimal values of the parameters, simply picked from literature. The latter can be rectified by calibrating the RANS model with experimental data from a flow interaction that is similar to the regime in which the calibrated model will be used. The relative importance of the two sources of errors is unknown.

Our interest lies in devising a principled way for calibrating (also called tuning) a RANS model for compressible, high Reynolds number jet-in-crossflow (JIC) interactions. JIC interactions are poorly simulated by RANS when using parameters held at their nominal values [10]. As a first step towards improving the predictive accuracy of RANS in JIC, we hypothesized that more relevant parameter values could be obtained by calibrating to a strongly vortical flow. In our previous work [11], we tested this hypothesis by designing a Bayesian calibration technique to estimate 3 RANS parameters, $\mathbf{C} = (C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$, using data from an *incompressible*, flow over a square cylinder (FOSC) experiment. The parameters were estimated as a 3D joint PDF (JPDF), capturing the uncertainty in the estimates due to limited data and the inherent shortcomings of RANS. A Markov chain Monte Carlo sampler was used to construct the PDFs (probability density functions) and its formidable computational cost was surmounted by using surrogate models of the RANS simulator. (A surrogate model is a statistical "curve-fit" which maps a RANS solver's outputs of interest to RANS parameters/inputs being varied or calibrated. Most often, they are constructed using polynomials and Gaussian Processes; see Ref. [12] for a comprehensive review of surrogate models. Surrogates are used as fast running proxies of expensive engineering and scientific models in applications such as optimization where they are invoked repeatedly.). The predictions obtained by running JIC simulations using parameters sampled from the post-calibration JPDF were far more accurate than the one obtained using $\mathbf{C}_{nom} = \{0.09, 1.92, 1.44\}$, the nominal values of the same parameters. The main contributions of our previous work were (1) to identify how inadequate $\mathbf{C}_{nom}$ was for JIC interactions, and a more relevant flow interaction that could be used to obtain better parameter settings $\mathbf{C}$; (2) to construct a rigorous method of creating surrogate models for Reynolds stresses in FOSC interactions; (3) to develop a JPDF for $\mathbf{C}$ using MCMC and surrogate models of a relatively complex flow (2D turbulent flow over a square cylinder) and (4) to demonstrate the usefulness of ensemble predictions. Using parameters sampled from the post-calibration PDF, we found a parameter set that was very close (about 15% relative errors) to measurements from a JIC experiment.

While our previous work showed the inadequacy of $\mathbf{C}_{nom}$ and a way to overcome it, it nevertheless did not address whether $k - \epsilon$ models for JIC could be improved further. The prediction errors in Ref. [11] contained contributions from the model-form error as well as parametric sub-optimality. In this study, we hypothesize that the bulk of the errors in RANS predictions of compressible JIC interactions are due to inappropriate parameters and can be addressed by proper calibration. We will calibrate to experimental data, obtained with a $M = 3.73$ jet exhausting into a $M = 0.8$ crossflow. Thereafter, any deviations of RANS predictions from observations can be attributed to model-form deficiencies of the $k - \epsilon$ model itself and the inaccuracies in the experimental data. We will also investigate whether the calibration is overly sensitive to the gross flow properties, specifically the strength of the jet and the freestream Mach number.

As a first step to calibrating to JIC experimental data, we attempted to create Gaussian Process (GP) and polynomial surrogate models that were accurate in the entire $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ space for JIC interactions i.e., in the manner described in Ref. [11]. It failed; the response of our calibration variable (vorticity in the cross-plane; described in detail in Sec. III) to variations in $\mathbf{C}$ is too complex and non-stationary (in the statistical sense i.e., the lengthscale of variation of the response changes in the parameter space) to be captured by simple surrogate models. Instead, we used experimental data to identify a promising, but irregular sub-domain in the parameter space, which had a higher potential to provide predictive RANS simulations. We call this region $\mathcal{R}$. We developed polynomial surrogates that are valid only inside $\mathcal{R}$. We use $\mathcal{R}$ as an improper prior (for Bayesian calibration), with uniform density inside $\mathcal{R}$, and zero outside. We develop a support vector machine classifier (SVMC) to model $\mathcal{R}$ and use that inside an MCMC calibration of $\mathbf{C}$. The use of a classifier to model the valid parameter region $\mathcal{R}$ is borrowed from climate modeling [13]; however, surrogate models which are valid only in a "sensible" portion of the parameter space have been used in aerospace design optimization [14]. The use of surrogate models that are valid in only a portion of the parameter space and a classifier that acts as a prior on the valid parameter subspace is novel in Bayesian parameter estimation and is the second contribution of this work.

The paper is structured as follows. In Sec. II, we review literature on calibration of aerodynamic models (both Bayesian and otherwise) and the use of surrogate models for calibration purposes in aerodynamics. We also review support vector machine classifiers. We describe relevant work in JIC simulations. In Sec. III, we formulate our inverse problem for estimating $\mathbf{C}$ (it is a conventional one), identify the need to develop surrogate models, and provide the details of the construction. We also describe in detail the experimental studies that supply us with our calibration data. In Sec. IV, we provide results of the calibration, and check the predictive skill of the calibrated RANS model for variables omitted from the calibration's likelihood. We also investigate the usefulness of the calibration at off-calibration points. We conclude in Sec. V.

II. Background

A. Jet-in-crossflow interactions

JIC is a canonical flow interaction and is used as a model for a host of practical and natural flow phenomena [15]. It also plays a central role in the maneuvering of certain finned slender aerodynamic bodies e.g., launch vehicles by spin rockets. In such cases, the exhaust from the spin rockets may interact with control surfaces, modifying aerodynamic forces and moments [16]. The problem is strongly vortical. Our interest lies in JIC interactions where the freestream (or crossflow) is transonic and the jet supersonic, a problem that has been investigated experimentally [17–20]; we will refer to these as the "Beresh experiments". The jet undergoes a Kelvin-Helmholtz roll-up of the shear layer formed at the boundary of the jet and crossflow, and the flowfield is dominated by a counter-rotating vortex pair (CVP). The CVP tracks the evolution of the jet in the crossflow, and has horseshoe vortices circling it. The CVP and the horseshoe vortices are primarily responsible for modification of the flow in the vicinity of the control surfaces.

The problem of JIC, in general, has also been studied numerically; references can be found in the review paper by Mahesh [15]. The particular supersonic-jet-in-transonic-crossflow that is the subject of this study has been investigated using $k - \omega$ models [10], which were compared with the Beresh experiments. The study found that all $k - \omega$ models overpredicted turbulent intensities inside the jet, resulting in simulated jets "fatter" than their experimental counterpart i.e., the turbulent diffusion was too large in the simulations. Further, the CVP, as captured on the crossplane (a plane perpendicular to the streamwise direction, that slices through the CVP) resided at a point higher than in the experiments, a result at odds with overly strong turbulent diffusion. The study conjectured that turbulent stresses were underpredicted in the jet's nearfield, resulting in an erroneous exchange of momentum between the jet and the crossflow. Large Eddy Simulations [21, 22] (LES) and Detached Eddy Simulations [23] (DES) of the same JIC configuration have also been performed and compared to the Beresh experiments; the mean-flow on the midplane was used as the figure of merit. No comparisons were performed on the crossplane. As expected, their agreement with experimental observations were far better than RANS. While this is encouraging, DES and LES are too computationally intensive to be used in routine design calculations. Thus the need to analyse and reduce RANS prediction errors retains its importance.

B. The RANS $k-\epsilon$ model

We employ a compressibility-corrected $k-\epsilon$ RANS model [24, 25]. Our calibration parameters $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ appear in the equations modeling turbulent kinetic energy k and its dissipation rate ϵ

$$\begin{aligned}
\frac{\partial}{\partial t}\overline{\rho} k + \frac{\partial}{\partial x_j}\overline{\rho}\tilde{u}_j k &= \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_T}{\sigma_k}\right)\frac{\partial}{\partial x_j} k\right] + P_k - \overline{\rho}\epsilon + SS_k \\
\frac{\partial}{\partial t}\overline{\rho}\epsilon + \frac{\partial}{\partial x_j}\overline{\rho}\tilde{u}_j \epsilon &= \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_T}{\sigma_\epsilon}\right)\frac{\partial}{\partial x_j}\epsilon\right] + f_1 C_{\epsilon 1} P_k \frac{\epsilon}{k} - f_2 C_{\epsilon 2}\overline{\rho}\hat{\epsilon}\frac{\epsilon}{k}
\end{aligned} \tag{1}$$

where

$$\begin{aligned}
P_k &= \overline{\rho}\tau_{ij}\frac{\partial}{\partial x_j}\tilde{u} \\
\mu_T &= f_\mu C_\mu \overline{\rho}\frac{k^2}{\epsilon} \\
\overline{\rho}\tau_{ij} &= \mu_T\left(\frac{\partial}{\partial x_j}\tilde{u}_i + \frac{\partial}{\partial x_i}\tilde{u}_j\right) - \frac{2}{3}\delta_{ij}\left(\mu_T\frac{\partial}{\partial x_n}\tilde{u}_n + \overline{\rho} k\right) \\
\hat{\epsilon} &= \epsilon - \frac{14}{9}\frac{\mu}{\overline{\rho} f_2}\frac{\partial}{\partial x_j}\sqrt{k}\frac{\partial}{\partial x_j}\sqrt{k} \\
SS_k &= \alpha_1 \hat{M}_T^2 P_k - \alpha_2 \hat{M}_T^2 \overline{\rho}\epsilon \\
\hat{M}_T &= \max(M_T - 0.2, 0)
\end{aligned} \tag{2}$$

Here, the damping functions f_1, f_2 and f_μ assume values significantly different from 1 (and tending towards zero) near the wall. $M_T = \sqrt{2k}/c$ is the turbulent Mach number. SS_k is the compressibility correction for the turbulent kinetic energy. The nominal values of the closure parameters are:

$$\begin{aligned}
&C_\mu = 0.09 \;\; C_{\epsilon 2} = 1.9 \;\; C_{\epsilon 1} = 1.43 \;\; \sigma_k = 1 \\
&\sigma_\epsilon = 1.0 \quad\;\; \alpha_1 = 2.5 \quad \alpha_2 = 2.0
\end{aligned}$$

As described in Ref. [26], the parameter values are estimated from simple, canonical, incompressible flows and it is unclear why they should be expected to be predictive in transonic and supersonic regimes. The standard value of $C_{\epsilon 2} = 1.92$ is obtained from incompressible decaying turbulence, though RNG $k-\epsilon$ models have it as 1.68 and experimental data [27] suggests it is 1.77. C_μ is estimated by setting $P_k - \overline{\rho}\epsilon = 0$, obtained in free shear flows. It yields $C_\mu = (\overline{u_1' u_2'}/k)^2 \sim 0.09$ from DNS data [28]. In uniform shear flows ($\partial u_1/\partial x_2 =$ constant), the following relation holds

$$\frac{P_k}{\epsilon} = \frac{C_{\epsilon 2} - 1}{C_{\epsilon 1} - 1} = 2.09,$$

leading to $C_{\epsilon 1} = 1.43$. However, experimental investigations [29] find that the ratio varies between 1.33 and 1.75, rather than 2.09. σ_k is set to 1, though there is no experimental justification for it [30], and RNG $k-\epsilon$ models use 0.72. Fully developed turbulent channel flows, where $Dk/Dt = D\epsilon/Dt = 0$ and $\kappa^2 = \sigma_\epsilon \sqrt{C_\mu}(C_{\epsilon 2} - C_{\epsilon 1})$, where $\kappa = 0.41$ is the von Karman constant, is used to estimate σ_ϵ. Thus even in the context of incompressible flows, the nominal values of the closure parameters introduce inconsistencies because of the tendency to treat them as "universal" constants that can be estimated from simple flows. In reality, they are tunable constants, and their optimal values are flow-dependent.

C. Improving the predictive skill of RANS models

The accuracy that LES and DNS (Direct Numerical Simulations) simulations can achieve have allowed them to be used as "numerical experiments" to quantify RANS errors as well as to calibrate RANS models in simple flow configurations. Both types of studies require one to augment the empirical models in RANS for the creation and dissipation of turbulent kinetic energy with a spatially-variable "correction" or model-form error term, which is estimated from LES or DNS data. The estimation of the model-form error has been investigated in three ways. In the first case (typically geared towards uncertainty quantification of RANS predictions), the parameters of the RANS model are deemed accurate, and the discrepancy between RANS and LES (or DNS) is computed. In Ref. [31], turbulence variables from 2D RANS and LES simulations of transonic flows over a bump were compared to estimate a spatially variable error. The error was then modeled spatially and included a random term to evaluate its impact on flow variables. This error model was tested by performing an ensemble of RANS simulations and adding realizations of the error field by drawing from the PDF. Recently, this approach has been extended to 3D JIC computations [32].

The second approach is, quite simply, the calibration of RANS models for a specific class of flows. Such studies have been performed for simple flows e.g., channel flows, flow over a flat plate etc. [33–35], and involve estimating both flow turbulence model parameters via inverse modeling as well as fitting a structural error model. The parameters are estimated as PDFs. In Ref. [34], parameter estimates and model-form error are estimated for the Spalart-Allmaras model using a Bayesian approach, with observational data from incompressible boundary layer flows over a flat plate under both favorable and adverse pressure gradients. The study by Edeling *et al.* [26] addressed the same problem, but used a $k-\epsilon$ model instead. A subsequent study by Edeling *et al.* [36] calibrated a number of RANS models ($k-\omega$, $k-\epsilon$, Spalart-Allmaras etc.) to flat-plate boundary layer data (under various pressure gradients) and combined them, via Bayesian Model Averaging, into a form that was predictive for pressure gradients not included in their calibration data.

The third approach for improving RANS models ascribes the lack of predictive skill of RANS models entirely to model-form error. As described in [37], the model-form error is represented as a spatially dependent source field in a key turbulent transport equation e.g., the equation for turbulent viscosity in Spalart-Allmaras models or the equation for k in $k-\omega$ models. This source field is estimated from LES data via adjoint-based gradient descent algorithms. The source field so obtained is then related to local flow variables via machine learning methods. A dependence on a large number of local flow variables is imputed and linked to the source term via a neural network; a hill-climbing algorithm is used to eliminate irrelevant local flow variables. The neural net model of the source field so constructed is then added to the RANS model and used in simulations. Ref. [38] describes the machine learning algorithm in detail and shows examples of source terms, trained on flat-plate and channel flows, being used in aerofoil simulations. Their later work [39] investigated the use of Multiscale Gaussian Processes as an alternative to neural networks. In [40], the authors adopt a Bayesian formulation and compute the estimation error in their source field (using the Hessian in their deterministic field inversion).

Some of the studies described above i.e., Refs. [26, 34, 36] used MCMC to perform the calibration and all inferences were probabilistic. The simplicity of the flow configuration (incompressible, flat-plate boundary layer) was instrumental in reducing computational costs and making the phenomenally expensive MCMC method feasible for calibration purposes. The use of MCMC to calibrate models for more complex flows requires one to replace the flow simulator with a surrogate. In Ref. [41] the authors constructed surrogate models, using kriging, for the velocities and temperatures predicted by LES in the crossplane of a hot jet-in-crossflow interaction. They used the surrogates to estimate optimal values of the constant in the Smagorinsky model and a fourth-order dissipation constant using JIC data obtained from a DNS simulation. Our previous work on RANS calibration [11] using FOSC data was discussed in Sec. I. In [42], the authors performed a Bayesian calibration of 4 $k - \epsilon$ parameters using measurements of turbulent kinetic energy (at 10 locations) from an experiment of urban canyon flow. Similar to the approach adopted here, they constructed a Gaussian Process surrogate of the 3D RANS simulator and inferred PDFs of the parameters using MCMC. Estimates of the numerical discretization error, model-form error (called bias in their work) and measurement error were also modeled and estimated as PDFs. The surrogate was valid only in a part of the parameter space where $C_{\epsilon 2} > C_{\epsilon 1}$, a feature that mirrors ours. They checked their calibration by reproducing the observations; they did not check whether the calibration improved predictions of other flow quantities. Our work is perhaps closest to [42] in approach and spirit, though we target very different flow regimes in Mach number, Reynolds number and geometry. Further, in contrast to [42], we check the improvement in post-calibration predictive skill for flow variables not included as observations, for the calibration and two off-calibration interactions.

D. Support vector machine classifier

The $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ parameter space for RANS models is defined using upper and lower bounds on each parameter. A random combination of parameters may not even be physical i.e., the RANS model may not even run successfully to completion. Further, non-physical parameter combinations do not provide model predictions that are close to experimental observations. In such cases, we can pragmatically excise the non-physical portion of the parameter space, construct a surrogate model that is valid only in $\mathcal{R}$, the physically realistic part of the parameter space, and use a classifier (a demarcation surface) to enforce $\mathcal{R}$ during the calibration process. One way to construct a demarcation surface between physical and non-physical parts of the parameter space is to define a function $\zeta(\mathbf{C})$ that assumes a value of ± 1 in the physical and non-physical parts respectively. Then the level-set $\zeta(\mathbf{C}) = 0$ defines the demarcation surface. The problem reduces to fitting a model for $\zeta(\mathbf{C}) = 0$. Support vector machines are a widely used "textbook" method for constructing such a classifier.

Consider a set of points $\{\mathbf{C}_i, \zeta_i\}, i = 1 \ldots N, \zeta_i = \pm 1$ and $\mathbf{C}_i \in \mathbb{R}^n$. If the points are separable i.e., points with the same ζ value cluster in a well-defined region of $\mathbf{C}$-space, then it may be possible to find a hyperplane L to act as a classifier. Let this hyperplane be given by $g(\mathbf{C}) = \mathbf{C}^T \beta / ||\beta||_2 + \beta_0$.

As described in [43], Chapter 4, the perpendicular distance from L to $\{\mathbf{C}_i, \zeta_i\}$ is given by $g(\mathbf{C})$. The sign of $g(\mathbf{C})$ indicates the side of L on which $\{\mathbf{C}_i, \zeta_i\}$ lies i.e., the direction of the perpendicular vector. Define the direction such that $\zeta_i(\mathbf{C}^T\beta/||\beta||_2 + \beta_0) \geq 0$ for any point that is correctly classified. A good classifier is one that will maximize this distance for all the points i.e.,

$$\underset{\beta,\beta_0,||\beta||_2=1}{\text{maximize}}\ M, \qquad \text{subject to } \zeta_i\left(\mathbf{C}_i^T\frac{\beta}{||\beta||_2} + \beta_0\right) \geq M, i = 1\ldots N,$$

or setting $||\beta||_2 = 1/M$

$$\underset{\beta,\beta_0}{\text{minimize}}\ ||\beta||_2, \qquad \text{subject to } \zeta_i\left(\mathbf{C}_i^T\beta + \beta_0\right) \geq 1, i = 1\ldots N.$$

Here $||\cdot||_2$ is the ℓ^2-norm of a vector. Methods to solve for $\hat{\beta}$, the optimal solution, are in Ref. [43]. It can be shown that $\hat{\beta}$ depends on a small set of support points $\{\mathbf{C}_i, \zeta_i\}$ and the hyperplane L is called the optimal separating hyperplane. In case the separating surface is more complicated and a hyperplane does not suffice, one uses a nonlinear function $h(\mathbf{C})$

$$\underset{\beta,\beta_0}{\text{minimize}}\ ||\beta||, \qquad \text{subject to } \zeta_i\left(h(\mathbf{C}_i)\beta + \beta_0\right) \geq 1, i = 1\ldots N.$$

Here $h(\mathbf{C})$ is a row vector containing features of $\mathbf{C}$ i.e., polynomial or nonlinear transforms of its elements that capture the nonlinear geometry of the separating surface. The precise nature of the transform $h(\mathbf{C})$ is neither required nor specified; the fitting algorithm only requires the inner product $\langle h(\mathbf{C}_a), h(\mathbf{C}_b)\rangle = K(\mathbf{C}_a, \mathbf{C}_b)$ called the kernel. Polynomials, radial basis functions (RBF) and neural nets are some common kernels. In our work, we will use a cubic kernel i.e. $K(\mathbf{C}_a, \mathbf{C}_b) = (1 + \langle \mathbf{C}_a, \mathbf{C}_b\rangle)^3$. Our experiments with linear kernels i.e., $K(\mathbf{C}_a, \mathbf{C}_b) = \langle \mathbf{C}_a, \mathbf{C}_b\rangle$ were not succesful and the RBF kernel, $K(\mathbf{C}_a, \mathbf{C}_b) = \exp\left[-(\mathbf{C}_a - \mathbf{C}_b)^2\right]$, had a slightly higher misclassification rate than the cubic kernel. A classifier that can accommodate nonlinear separating surfaces using kernels is called a SVMC (support vector machine classifier); see [43], Chapter 12 for details.

III. Formulation of the calibration problem

In this section, we formulate a Bayesian inverse problem for estimating $\mathbf{C} = \{C_\mu, C_{\epsilon 2}, C_{\epsilon 1}\}$ from experimental measurements. We identify the need for surrogate modeling and also develop an informative prior.

Let $\mathbf{y}_e$ be a vector (of length N_p) of experimental observations, measured at a set of N_p locations ("probes"). Let $\mathbf{y}_m(\mathbf{C})$ be model predictions of the same, produced by a parameter setting $\mathbf{C}$. They are related by $\mathbf{y}_e = \mathbf{y}_m(\mathbf{C}) + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon}$ is a combination of measurement and model-form error. We make a modeling assumption that the errors at the probes are uncorrelated, independently and identically distributed and can be modeled as draws from a zero-mean Gaussian i.e. $\boldsymbol{\epsilon} = \{\epsilon_i\}, \epsilon_i \sim \mathcal{N}(0, \sigma^2)$. σ^2 thus provides a crude measure of the model - data misfit after calibration. If the measurement errors are low, σ^2 provides an estimate of the model-form error.

Let $P(\mathbf{C}, \sigma^2 | \mathbf{y}_e)$ be the joint probability density function of the parameters and the model - data misfit, conditional on the observed data $\mathbf{y}_e$. Let $\Pi_1(\mathbf{C})$ and $\Pi_2(\sigma^2)$ be our prior belief regarding the distribution of $\mathbf{C}$ and σ^2. The likelihood of observing $\mathbf{y}_e$, given a parameter setting $\mathbf{C}$, $\mathcal{L}(\mathbf{y}_e|\mathbf{C})$, is given by

$$\mathcal{L}(\mathbf{y}_e|\mathbf{C}, \sigma^2) \propto \frac{1}{\sigma^{N_p}} \exp\left(-\frac{||\mathbf{y}_e - \mathbf{y}_m(\mathbf{C})||_2^2}{2\sigma^2}\right).$$

By Bayes' theorem, the calibrated distribution (or posterior distribution) of $(\mathbf{C}, \sigma^2)$ can be given as

$$P(\mathbf{C}, \sigma^2|\mathbf{y}_e) \propto \mathcal{L}(\mathbf{y}_e|\mathbf{C}, \sigma^2)\ \Pi_1(\mathbf{C})\ \Pi_2(\sigma^2) \propto \frac{1}{\sigma^{N_p}} \exp\left(-\frac{||\mathbf{y}_e - \mathbf{y}_m(\mathbf{C})||_2^2}{2\sigma^2}\right)\ \Pi_1(\mathbf{C})\ \Pi_2(\sigma^2) \quad (3)$$

This inverse problem can be solved via sampling i.e., we can use a method such as MCMC [44] to draw samples of $\{\mathbf{C}, \sigma^2\}$. We reconstruct $P(\mathbf{C}, \sigma^2|\mathbf{y}_e)$ empirically by plotting histograms or via kernel density estimation [45]. In this particular work, we use an adaptive MCMC method called Delayed Rejection Adaptive Metropolis (DRAM; Ref. [46]), as implemented in the R [47] package `FME` [48]. The convergence of the MCMC chain to a stationary distribution $P(\mathbf{C}, \sigma^2|\mathbf{y}_e)$ i.e., the determination of the sufficiency of $\{\mathbf{C}, \sigma^2\}$ samples to construct $P(\mathbf{C}, \sigma^2|\mathbf{y}_e)$, is performed using the Raftery-Lewis method [49] implemented in the R package `mcgibbsit` [50]. The prior density for σ^2 is defined in terms of its reciprocal i.e., $\Pi_2(\sigma^{-2})$ and is modeled using a Gamma prior i.e., $\sigma^{-2} \sim \Gamma(k, \theta)$, where $k = 1, \theta = 1$. The inverse Gamma prior for σ^2 is a conjugate prior which simplifies sampling of σ^{-2} via a Gibbs sampler. Also, the prior is virtually non-informative for $\sigma^2 > 5$.

The MCMC method requires $O(10^4)$ samples to construct $P(\mathbf{C}, \sigma^2|\mathbf{y}_e)$, each of which requires a 3D RANS model evaluation to provide $\mathbf{y}_m(\mathbf{C})$. Since this is impractical, we will develop a surrogate model, a polynomial that maps the dependence of our calibration variable (described below) on $\mathbf{C}$. This mapping is expected to hold true (i.e., within an acceptable level of error) in the support of $\Pi_1(\mathbf{C})$. The surrogate model will serve as a computationally inexpensive proxy for SIGMA CFD (Sandia Implicit Generalized Multi-Block Analysis Code for Fluid Dynamics), our compressible 3D RANS simulator. We consider the following bounds on $\mathbf{C}$ taken from Ref. [24]:

$$0.06 \leq C_\mu \leq 0.12, \quad 1.7 \leq C_{\epsilon 2} \leq 2.1, \quad \text{and} \quad 1.2 \leq C_{\epsilon 1} \leq 1.7. \tag{4}$$

While it is tempting to combine these bounds into a uniform distribution in the cuboid $\mathcal{C}$ in $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ space, we shall refrain from doing so since random samples drawn from it may not be physically relevant. While the experiment revealed that the flow is turbulent with steady mean-flow, many $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ combinations drawn from $\mathcal{C}$ lead to RANS computations that do not converge to steady state; other combinations lead to solutions that bear little resemblance to high Reynolds number transonic flowfields (and consequently the discrepancy $\mathbf{y}_e - \mathbf{y}_m(\mathbf{C})$ is very large). The inability to generate reasonable model predictions $\mathbf{y}_m$ makes it very difficult to construct surrogate models that are accurate proxies for RANS predictions everywhere in $\mathcal{C}$. Consequently, to circumvent this problem, we will develop a prior density that restricts the parameters to a region $\mathcal{R}$, within which the parameters produce physically realistic flowfields.

A. Experimental data

The wind tunnel experiments from which we obtain our calibration and validation data are described in Ref. [17–19]. The test section simulated using SIGMA CFD (see schematic in Fig. 1 (left)) has a square cross section, 304.8 mm per side. Turbulent flow is introduced into the test section from the left. A jet with diameter $D_j = 9.53$ mm and perpendicular to the inflow, is introduced at the floor of the test section at Mach 3.73. PIV measurements of the flowfield are conducted in the midplane (the plane of symmetry in JIC interaction) as well as a crossplane (perpendicular to the jet), 321.8 mm (33.8 jet diameters) downstream from the center of the jet. Measured quantities include velocities and the fluctuating turbulent velocities, from which turbulent stresses ($\tau_{ij} = -\overline{u'_i u'_j}$) are computed. The studies track the penetration of the jet into the crossflow. This tracking is performed by plotting streamwise and vertical velocity profiles at 5 locations,

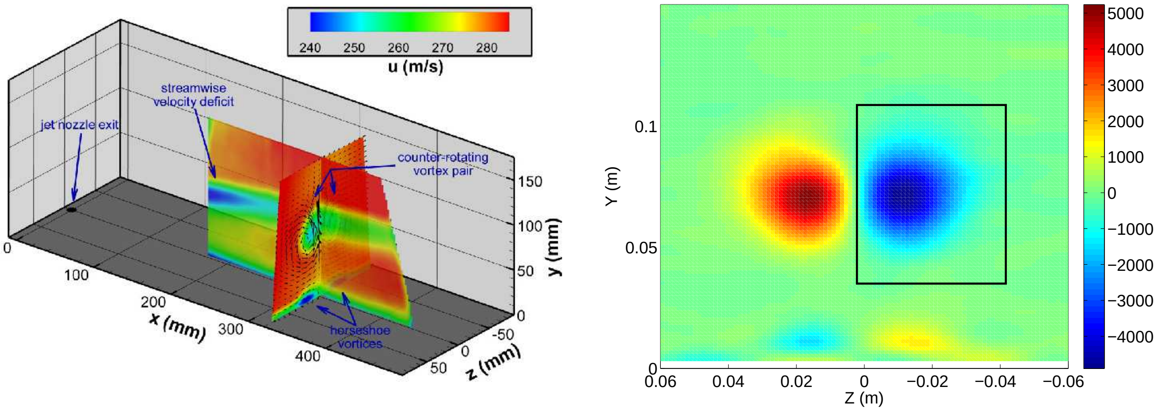

Fig. 1 Left: Schematic of the test section showing the orifice where the jet is introduced and the mid- and crossplane where experimental measurements are made. Right: Plot of the experimental streamwise vorticity and the window $\mathcal{W}$ ($0 \le z \le 0.04, 0.031 \le y \le 0.11$) containing one of the vortices. Calibration and validation based on vorticity will be performed inside this box.

starting at a distance 200 mm downstream of the center of the jet and thereafter, 50 mm apart. Velocity and turbulent stress distributions in the crossplane are used to demarcate the CVP. Most of the measurements are for $M = 0.8$ freestream flow, with $J = 10.2$. The wind-tunnel unit Reynolds number is $20 \times 10^6 \mathrm{m}^{-1}$. Midplane and crossplane measurements are also available for $J = 16.7$. Measurements are restricted to the midplane for $M = 0.7$.

In this study we will use the $M = 0.8, J = 10.2$ experiment as our calibration case. In particular, we will use the experimental velocity measurements on the crossplane to compute the streamwise vorticity. The calibrated model will be tested against a $M = 0.7, J = 10.2$ and $M = 0.8, J = 16.7$ JIC interactions. Fig. 1 (right) shows a plot of the experimental crossplane vorticity field, clearly showing the CVP. The window $\mathcal{W}$ ($0 \le z \le 0.04$ m$, 0.031$ m $\le y \le 0.11$ m), surrounding one of the vortices, demarcates the region where the vortices are observed in the RANS solution as $\mathbf{C}$ is varied. The window also eliminates the boundary layer seen at the bottom of the figure. Superimposing the RANS grid on the crossplane reveals that the window is covered by an 8×28 grid. We refer to these 224 grid cells as the set $\mathcal{P}_f$ of "probes" where both experimental and model predictions of streamwise vorticity are available for comparison and matching during calibration.

B. The RANS simulator

Our 3D, compressible JIC RANS simulations are performed using SIGMA CFD. SIGMA CFD implements a compressibility-corrected $k - \epsilon$ model [24] (outlined in Sec. II B) and uses a linear model to relate the strain rates to the turbulent stresses. The model uses turbulent Reynolds number based (y^+ independent) damping functions at the walls making it suitable for large scale parallel computations of complex flows. A Roe-TVD flux scheme with a minmod limiter for discontinuity capturing has been used for the spatial discretization. Time integration is carried out using a first order point-implicit scheme. The calculations are initialized using a first-order spatial scheme, time-marched for 5000 timesteps, and relaxed to convergence using the second order scheme for 25000 timesteps. Time marching to steady state was carried out using local time stepping with a gradual CFL ramp to accelerate convergence.

The simulations are performed on a multi-block mesh with approximately 10 million grid-cells. Earlier work [10] showed that mesh converged RANS results are obtained on this mesh. The simulation resolves the test section from 10 jet diameters upstream of the nozzle to about 60 jet diameters downstream of the nozzle. The near wall spacing is maintained at $y^+ = 1$ in this region. The jet nozzle is resolved from the stagnation chamber upstream of the throat, again using a mesh with near wall spacing at $y^+ = 1$. In order to provide inflow boundary conditions to the flow in the tunnel test section a full RANS simulation of the test section is carried out first. The profile at the station corresponding to the inlet boundary of the JIC domain is extracted and provided as boundary conditions to the JIC simulations. The details of the initialization (static and total pressures and temperatures for the jet and crossflow, etc.), necessary for reproducing experimental conditions are in [10, 17].

SIGMA CFD is primarily used to perform an ensemble of JIC simulations, corresponding to different $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ inputs, and construct a training data corpus for the surrogate model (see Sec. III C and III D). The values of σ_ϵ and σ_k are held at their nominal values [24], at variance with [42] where σ_k was estimated and

$$\sigma_\epsilon = \frac{\kappa^2}{\sqrt{C_\mu}(C_{\epsilon 2} - C_{\epsilon 1})}$$

was imposed. The expression is derived from fully developed turbulent channel flows where $Dk/Dt =$

$D\epsilon/Dt = 0$ and implies that $C_{\epsilon 2} > C_{\epsilon 1}$ to prevent σ_ϵ from becoming zero or negative. This constraint was imposed as a prior belief in [42]. Here, JIC is far from being a channel flow and the material derivatives are not zero, so it is not clear what prior belief should be employed. Hence we find it empirically in Sec. III C; our prior in Eq. 4 ensures $C_{\epsilon 2} > C_{\epsilon 1}$, and our empirical method also finds the margin $(C_{\epsilon 2} - C_{\epsilon 1})$ that must be maintained.

C. Designing an informative prior

Model calibration requires one to search for parameters which allow a model to reproduce experimental observations. While Eq. 4 establishes the bounds of the parameter space $\mathcal{C}$, arbitrary parameter combinations selected from this space may be non-physical, leading to unrealistic flowfields or may cause the RANS simulator to crash. In this section we devise a way of selecting $\mathcal{R}$, the part of $\mathcal{C}$ containing $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ values that produce physically realistic flowfields. We draw 2744 samples of $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ from $\mathcal{C}$ using a space-filling, quasi Monte Carlo method (Halton sequence, as implemented in the R package `randtoolbox` [51]) and use them to seed 3D RANS JIC simulations. Each simulation (approximately 12 hours on 1024 cores of a PowerPC A2 processor) results in a streamwise vorticity prediction on the crossplane. 100 runs failed and a number of the remaining 2644 runs resulted in non-physical flowfields. On the whole, the variation of $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ led to CVPs of varying sizes, strengths and locations.

In order to identify $\mathcal{R}$, we compute the root mean square error (RMSE) between the streamwise vorticity produced by each of the simulations and the experimental counterpart on the crossplane. We retain the $N_s = 525$ runs whose RMSE lie below the 20th percentile. The $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ for these selected runs, which identify $\mathcal{R}$, are plotted in Fig. 2. The 3D scatterplot clearly shows that $\mathcal{R}$ excludes a large portion of $\mathcal{C}$. Fig. 2, bottom right, clearly shows $C_{\epsilon 2} > C_{\epsilon 1}$. The threshold of 20th percentile was arrived at by trial and error. A larger percentile e.g., the median, resulted in a $\mathcal{R}$ that included the non-physical part of the parameter space. Its strongly non-stationary response surface could not be captured with a surrogate model (see Sec. III D). Setting a lower threshold e.g., 15th percentile resulted in a training set that was too small to construct surrogate models whose accuracy could be measured, via cross-validation, with any degree of confidence.

Our next step lies in using the training set of N_s $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ points to devise a function in $\mathcal{C}$

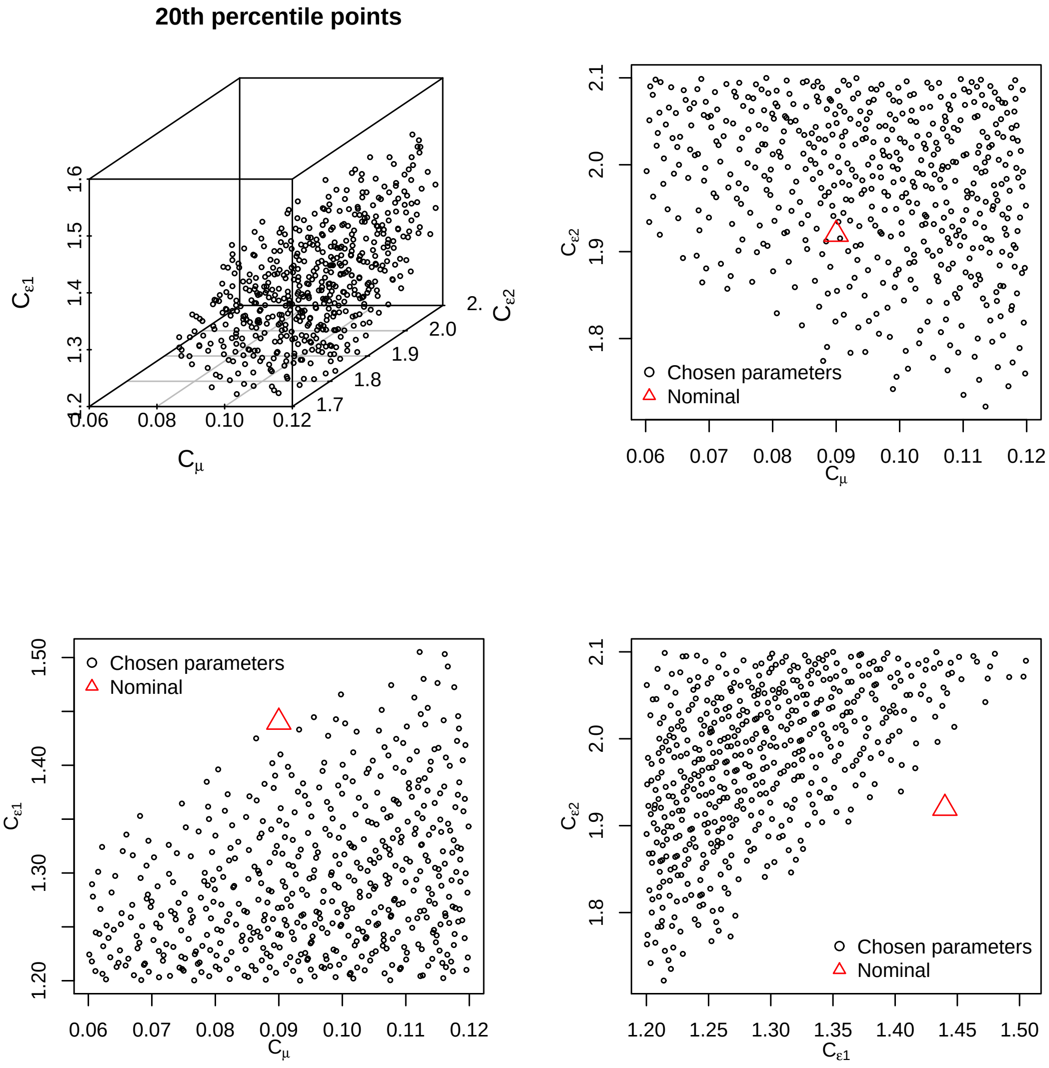

Fig. 2 Top left: Scatterplot of the set of N_s points in the $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ space that constitute $\mathcal{R}$. Top right: Projection of the points onto the $C_\mu - C_{\epsilon 2}$ plane. Bottom: Projections of the points on the $C_\mu - C_{\epsilon 1}$ and $C_{\epsilon 1} - C_{\epsilon 2}$ planes. The nominal value of $\mathbf{C}$, $\mathbf{C}_{nom}$ is also plotted.

that can be used to demarcate $\mathcal{R}$. We define a function $\zeta(\mathbf{C})$ in $\mathcal{C}$ that is set to 1 at the N_s points that define $\mathcal{R}$ and -1 at the remaining 2744 - 525 $=$ 2219 $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ combinations that we had drawn from $\mathcal{C}$. We define the level set $\zeta(\mathbf{C}) = 0$ as $\partial\mathcal{R}$, the separating surface for $\mathcal{R}$. We model $\partial\mathcal{R}$ using SVMC, with a cubic kernel. The SVMC is fitted using the R package `e1071` [52]. Five-fold cross-validation is used to estimate the hyperparameters of the SVMC. We test the accuracy of $\zeta_m(\mathbf{C})$ using repeated random subsampling validation (a form of cross-validation). We randomly divide the 2744 samples of $\zeta(\mathbf{C})$ into a Learning Set (LS) with 2332 samples (85% of the total) and a Testing Set (TS) with the rest. A SVMC is built using the LS and used to classify/predict at the $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ combinations in the TS. The misclassification rate is then computed. This is repeated 50 times, with different LS/TS pairs, and the mean misclassification rate over 50 rounds of cross-validation is found to be around 1%. Thus we find the resulting classifier to be quite accurate. We model the prior density within $\mathcal{R}$ as a uniform distribution i.e., all $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ combinations are equally probable. Outside $\mathcal{R}$, the prior density is zero.

D. Polynomial surrogate models

Let $y^{(p)}(\mathbf{C})$ be the streamwise vorticity on the crossplane, predicted by RANS at probe $p, p \in \mathcal{P}_f$. We seek to construct an approximation $y_s^{(p)}(\mathbf{C})$ using multivariate polynomials involving $\mathbf{C}, \mathbf{C} \in \mathcal{R}$. Since $y^{(p)}(\mathbf{C})$ is obtained by taking the derivative of the velocity field, it tends to be contaminated with some numerical noise. For probes p in or near the center of the CVP, the large magnitude of vorticity allows its dependence on $\mathbf{C}$ to be discovered and modeled. The rest of the probes, where numerical noise is substantial, have to be eliminated.

For a given RANS run, we compute the 75th percentile of the vorticity magnitudes obtained at probes $p, p \in \mathcal{P}_f$ and retain the $224/4 = 56$ probes with absolute vorticities above this threshold. The vorticities predicted at these probes are expected to be large enough to be minimally affected by numerical noise. This is repeated for all $\mathbf{C}$ in our training data of $N_s = 525$ RANS runs that lie inside $\mathcal{R}$; a different set of probes is selected for each run since the CVP changes its size and location as $\mathbf{C}$ varies. We take a union of these probes to create a set $\mathcal{P}_*$, containing 108 probes, where we will attempt to create surrogate models. The probes in $\mathcal{P}_*$ are plotted on top of the experimental vorticity field in Fig. 3 (left) with circles. Note that $\mathcal{P}_*$ is not the set of probes that we will use in the calibration; as we will see below, we will be able to create surrogate models, of an acceptable quality, only for a subset of them.

The method to create and test the robustness of surrogate models is the same as the one

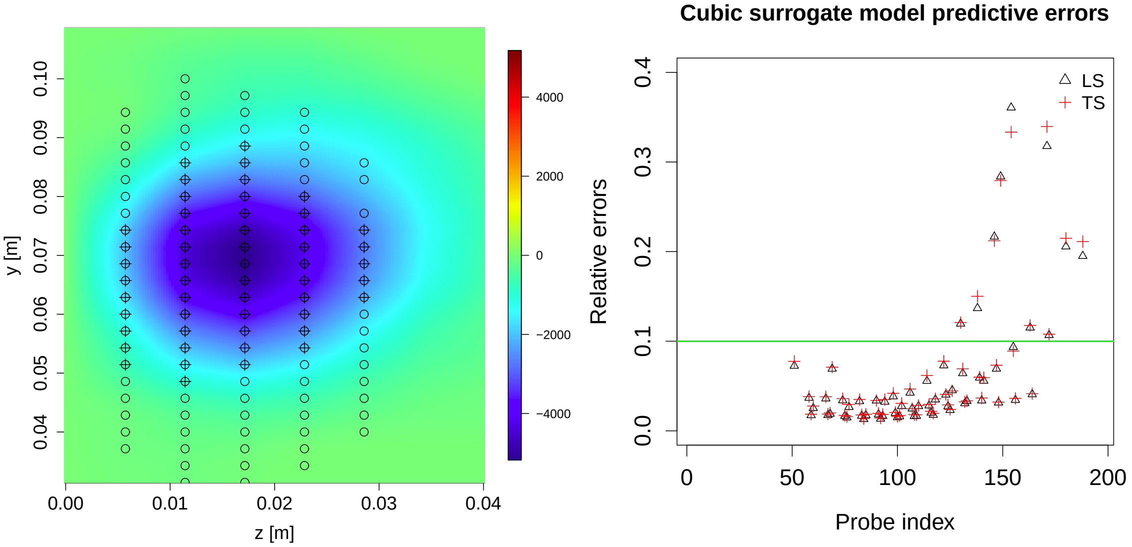

Fig. 3 Left: Plot of experimental vorticity inside $\mathcal{W}$ with the probes constituting $\mathcal{P}_*$ overlaid on it and marked with $\circ$. Right: Comparison of $\overline{\delta^{(p)}_{LS}}$ and $\overline{\delta^{(p)}_{TS}}$ for all 108 probes in $\mathcal{P}_*$. We note that the two relative errors are of the same magnitudes, indicating that there is little overfitting of the surrogates. We also see that for some probes, the relative errors can be big i.e., surrogates are not necessarily accurate for all probes. Some of the errors also lie outside the range of the vertical axis and are not plotted. The horizontal line at 10% error demarcates the probes that can be used in calibration; the rest of the polynomial surrogates are too inaccurate for any practical use. The probes which lie under the green line are plotted on the right with + symbols.

described in Ref. [11]. At any probe $p, p \in \mathcal{P}_*$, we postulate that

$$\begin{aligned} y^{(p)} &= \sum_{l=0}^{l=3} \sum_{m=0}^{m=3} \sum_{n=0}^{n=3} \alpha^{(p)}_{l,m,n} (C_\mu)^l (C_{\epsilon 2})^m (C_{\epsilon 1})^n + d, \\ &= y^{(p)}_s + d, \\ 3 &\geq l + m + n, \end{aligned} \tag{5}$$

where $y^{(p)}_s$ is the polynomial approximation (surrogate) of the 3D RANS prediction of the crossplane streamwise vorticity $y^{(p)}$ and d is the approximation error. In order to estimate the coefficients $\alpha^{(p)}_{l,m,n}$, we use the dataset $\{C_\mu, C_{\epsilon 2}, C_{\epsilon 1}, y^{(p)}\}_i, \mathbf{C}_i \in \mathcal{R}, i = 1 \dots 525$, and fit the model in Eq. 5 via least-squares fitting. We simplify the model by incrementally removing high-order terms and refitting the shrunk model to the same data. The (original) cubic model and its shrunk counterpart are competed using AIC (Akaike information criterion); if the shrunk model results in a lower AIC, it is retained and subjected to the same incremental simplification process. We find that simplification removes terms from Eq. 5 for all probes in $\mathcal{P}_*$, and in some cases reduces the cubic model to quadratic.

We next test whether the AIC-based simplification procedure yields robust surrogate models. This is done via repeated random subsampling validation, and results are averaged over 100 rounds. The N_s RANS runs are separated into an LS containing 446 (approximately 85% of the runs) randomly selected runs, while the remainder constitute the TS. The simplified polynomial model is constructed using the LS and the relative error for each parameter combination i in the LS, $\delta^{(p)}_{i,LS} = (y^{(p)}_i - y^{(p)}_{s,i})/y^{(p)}_i$, $i \in$ LS, is evaluated. Here $y^{(p)}_i$ refers to RANS predictions computed using the parameters $\mathbf{C}_i, i \in LS$. The fitted model is also used to evaluate $y^{(p)}_{s,j}$ for the parameter set $j, j \in TS$, and calculate the corresponding relative error i.e., $\delta^{(p)}_{j,TS} = (y^{(p)}_j - y^{(p)}_{s,j})/y^{(p)}_j$, $j \in$ TS. These individual relative errors are summarized by their RMS (root mean square) value for the LS and TS. This process is repeated 100 times, using different LS/TS pairs; the means of the RMS relative errors, $\overline{\delta^{(p)}_{LS}}$ and $\overline{\delta^{(p)}_{TS}}$, are taken as measures of accuracy of the polynomial surrogate. The process is repeated for all the 108 probes in $\mathcal{P}_*$.

In Fig. 3 (right) we plot $\overline{\delta^{(p)}_{LS}}$ and $\overline{\delta^{(p)}_{TS}}$ for all the probes. We notice that for about half the probes, the two relative errors are very close i.e., the polynomial model fitted to the LS data is equally predictive of the TS. This indicates that the polynomial model does not overfit the LS data; had this not been the case, $\overline{\delta^{(p)}_{LS}}$ would have been substantially smaller than $\overline{\delta^{(p)}_{TS}}$. We also notice that for certain probes the relative error is high i.e., the polynomial model is not an accurate representation of $y^{(p)}$, and should not be used in calibration. Using 10% error (horizontal line in Fig. 3 (right)) as the criterion, we retain 52 out of 108 probes in $\mathcal{P}_*$ where the polynomial model is deemed to be sufficiently accurate. Further, for these selected probes, the LS and TS relative errors are very close. We refer to this set of 52 probes as $\mathcal{P}$; since we can construct accurate surrogate models for all probes $p, p \in \mathcal{P}$, we will use them for calibration purposes. The probes in $\mathcal{P}$ are plotted in Fig. 3 (left) using + signs. They occupy the center of the experimental vorticity field with large vorticity magnitudes.

We also tested Gaussian Process (GP) surrogates for representing vorticity as a function of $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$. They were about as accurate as polynomial surrogates inside $\mathcal{R}$, but far more computationally expensive. Also, since our GP surrogate was based on a stationary covariance matrix, it did not perform well in the non-physical parts of the parameter space, where the response of vorticity to $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ was strongly non-stationary.

E. The inverse problem

We will solve a slight variation of Eq. 3 to estimate $\mathbf{C}$ from the Beresh experiments

$$P(\mathbf{C}, \sigma^2|\mathbf{y}_e) \propto \mathcal{L}(\mathbf{y}_e|\mathbf{C}, \sigma^2)\ \Pi_1(\mathbf{C})\ \Pi_2(\sigma^2) \propto \frac{1}{\sigma^{N_p}} \exp\left(-\frac{||\mathbf{y}_e - \mathbf{y}_s(\mathbf{C})||_2^2}{2\sigma^2}\right)\ \Pi_1(\mathbf{C})\ \Pi_2(\sigma^2), \qquad (6)$$

the modification being the substitution of crossplane streamwise vorticity predictions via a 3D RANS model, $\mathbf{y}_m$, by predictions via surrogates $\mathbf{y}_s = \{y_s^{(p)}\}, p \in \mathcal{P}$. Post-calibration, we will draw samples of $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ from $P(\mathbf{C}, \sigma^2|\mathbf{y}_e)$ and run RANS simulations to generate an ensemble of crossplane vorticity predictions. Statistical summaries of the vorticity field lying inside the window $\mathcal{W}$ will be used for validation purposes. We will also check the degree of accuracy with which we can predict the mean flow on the midplane. This will be performed for three cases: $M = 0.8, J = 10.2$ (the calibration case), $M = 0.8, J = 16.7$ and $M = 0.7, J = 10.2$ (the off-calibration cases).

IV. Results

We first check our software infrastructure for solving Eq. 6 to develop a posterior distribution for $\mathbf{C}$. We do so by performing a "synthetic data inversion", where we construct the "observations" (vorticity on the crossplane) using $\mathbf{C}_{synth} = \{0.097, 2.06, 1.37\}$. These "observations" were generated using the RANS simulator and will differ from surrogate model predictions by about 5%, the surrogate modeling error; see Fig. 3 (right). We then calibrate $\mathbf{C}$ using these synthetic observations and surrogate models (note: $\mathbf{C}_{synth}$ is not included in our training data for surrogate models); we expect that our PDFs will peak at $\mathbf{C}_{synth}$ if the software infrastructure is correct. Fig. 4 plots the marginalized posterior PDFs from the synthetic data case. Comparing the posteriors (solid line) to the priors (dashed line), we see a reduction in parametric uncertainty, especially for $C_{\epsilon 2}$ and $C_{\epsilon 1}$. We also see that the PDFs peak at the true parameter value $\mathbf{C}_{synth}$. The width of the PDFs (uncertainty in the estimate of $\mathbf{C}_{synth}$) is due to limited observations (data at 52 probes constituting

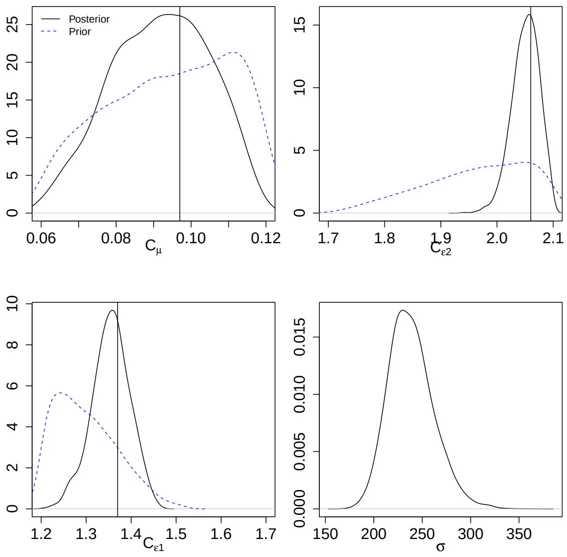

Fig. 4 Marginalized PDFs for $C_\mu, C_{\epsilon 2}, C_{\epsilon 1}$ and σ, computed as a solution to Eq. 6, using synthetic data. The true values of the parameters are shown using the vertical line. The dashed line is the prior i.e., $\mathcal{R}$. We see that the calibration significantly shrinks the uncertainty of the parameters.

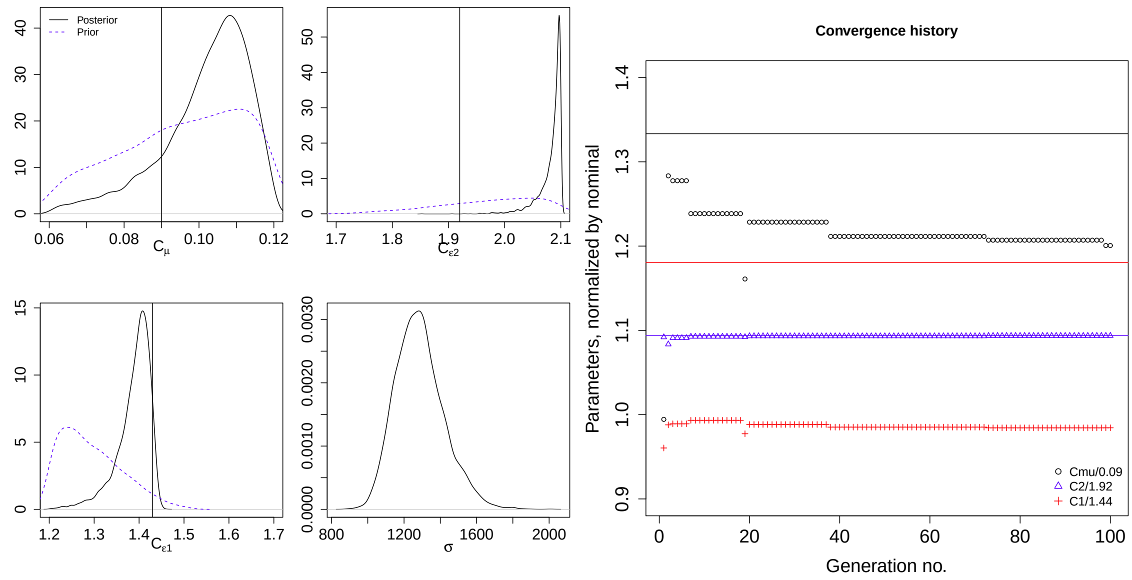

Fig. 5 Left: Marginalized PDFs for $C_\mu, C_{\epsilon 2}, C_{\epsilon 1}$ and σ computed as a solution to Eq. 6. The nominal values of the parameters are shown using the vertical line. The dashed line is the prior i.e., $\mathcal{R}$. The MAP estimates of all parameters except $C_{\epsilon 1}$ vary significantly from their nominal value. Right: Convergence plot of the GA estimates of C. The solid lines indicate the upper bounds of the 3 parameters.

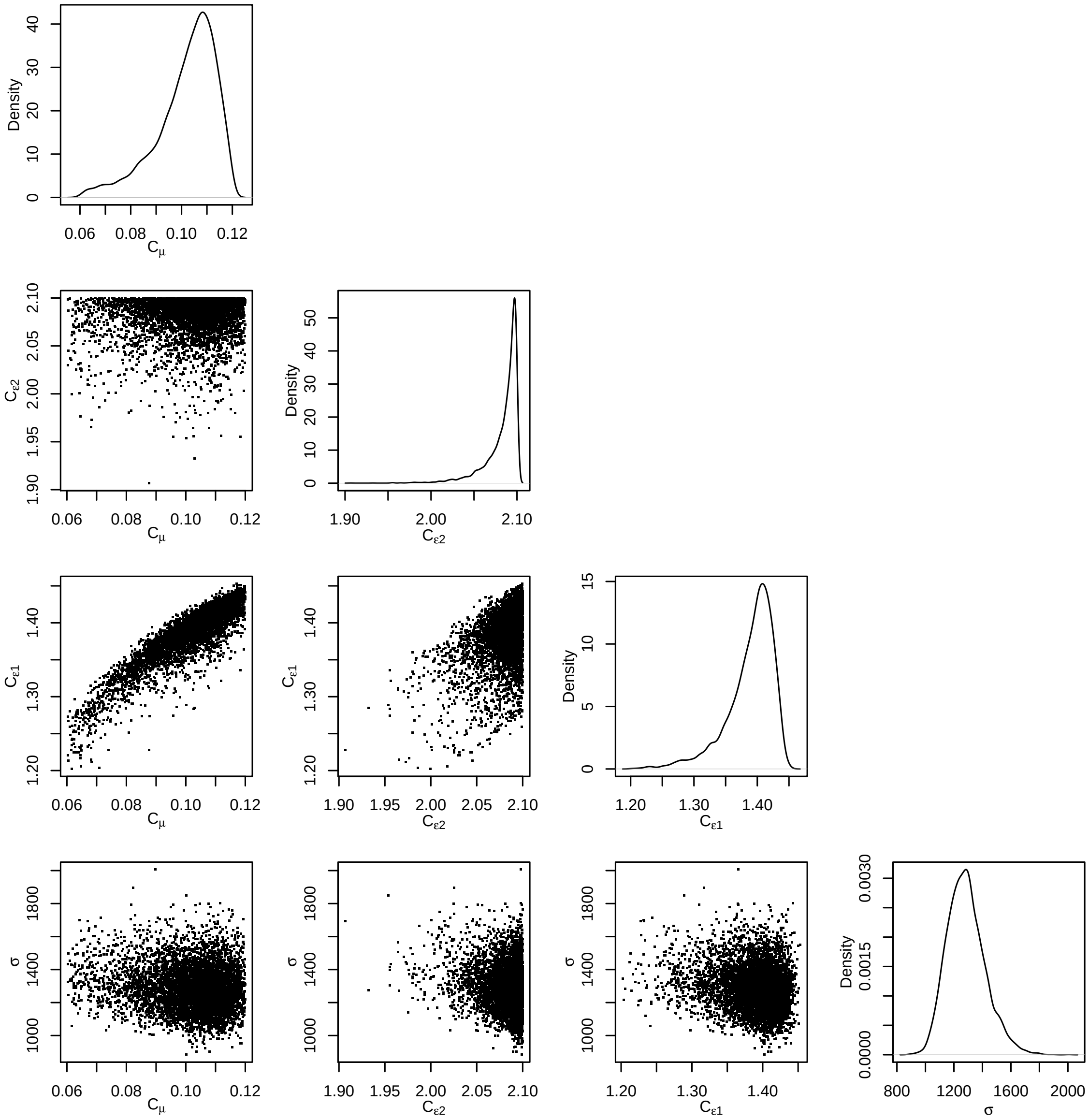

Fig. 6 Two-dimensional marginalizations of the posterior distribution. The correlations between the components of $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ **are clearly seen.**

$\mathcal{P}$) and the approximation error between RANS and the surrogate models. The slight estimation bias in $C_{\epsilon 1}$ (the shift between the true and the *maximum a posteriori* or MAP value or the "peak of the PDF" value of the parameter) is due to the same reason. These parameter estimates do not suffer from measurement or model-form errors, and the estimate for σ in Fig. 4 assesses only the impact of surrogate model errors.

The Bayesian inverse problem defined in Eq. 6 requires about 28,000 MCMC steps to reach a stationary posterior distribution. Analyses of the MCMC chains and tests to determine convergence to a stationary 4-dimensional ($\{C_\mu, C_{\epsilon 2}, C_{\epsilon 1}, \sigma^2\}$) distribution are in the Appendix A. The posterior is marginalized and the 1D PDFs for the individual parameters are plotted in Fig. 5 (left); the nominal values are plotted using vertical lines. It is clear that the MAP values of the parameters vary significantly from the nominal ones, except for $C_{\epsilon 1}$. Our calibration favors values of C_μ larger than the nominal, in contrast to the findings in Ref. [42]. The MAP value of $C_{\epsilon 2}$ obtained is similar to [42], though its proximity to the upper bound shows that the bounds in Ref. [42] are more appropriate. The MAP estimate of $C_{\epsilon 1}$ is similar to the nominal one, and agrees with the one found in Ref. [42]. Note that while the MAP values may agree, the PDFs inferred in our study are not at all similar to those in [42]; given the difference in compressibility, Reynolds number (50,000 in [42]) and flow geometry, this is not entirely surprising. An estimate of the model - data mismatch in the form of σ is also obtained. Comparing σ with the vorticity plotted in Fig. 3 (left), we see that the misfit is quite large. Further, comparing the plots for σ in Fig. 4 and Fig. 5, we see that σ, as estimated from experimental data (MAP value of around 1400), is about six times larger compared to the synthetic data case (MAP value around 250). The former assesses the effect of measurement, model-form and surrogate modeling error, whereas the latter is limited only to surrogate modeling.

As a check, we also performed a calibration using the same formulation, but with a genetic algorithm as the global optimizer. The optimization is performed with surrogate models, using the genetic algorithm implementation in the R package `GA` [53]. GA algorithms are described in [54]. We start with a population of 200 parameter settings $\mathbf{C}$ drawn from $\mathcal{R}$, and run for 100 iterations. At each iteration each ensemble member could be crossed over with another with probability 0.8, and could undergo a mutation with probability 0.33. Mutations were uniform random mutations in the components of $\mathbf{C}$, whereas crossovers were blended. Mutations and crossovers produce offspring of different fitness (i.e., an ability to reproduce observations) and increase the population. At the end of the iteration, the population is thinned; members are selected with a probability linearly proportional to their fitness. The convergence of the algorithm is in Fig. 5 (right). The best/converged value of the parameters are $C_\mu = 0.105, C_{\epsilon 2} = 2.099, C_{\epsilon 1} = 1.42$. In Fig. 6, we plot 2-dimensional marginalizations of the JPDF. We see significant correlations between $C_{\epsilon 1}$ and the

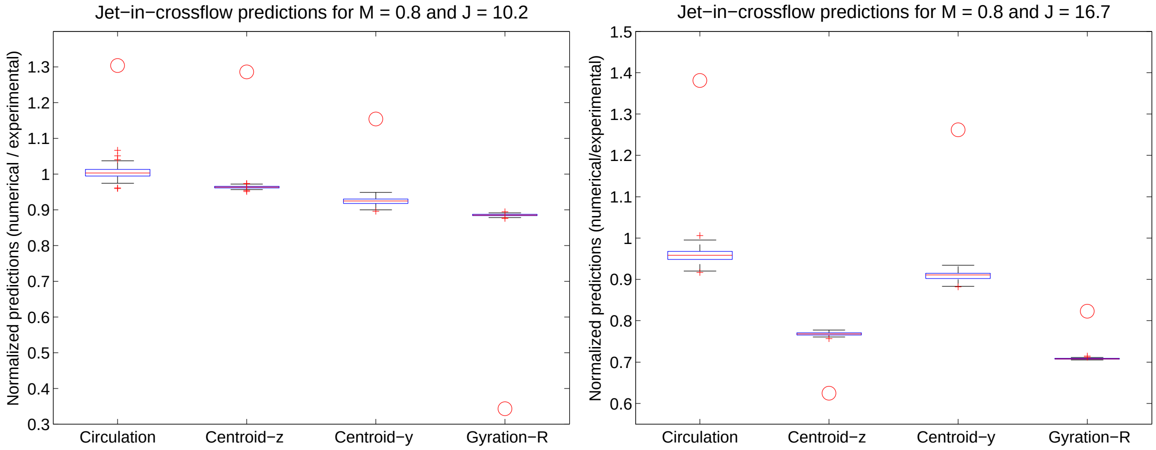

Fig. 7 Left: Distribution of circulations, the centroids and radii of gyration generated from 100 samples from the posterior distribution, normalized by their experimental counterparts. Predictions using $\mathbf{C}_{nom}$ are plotted with ○. The results correspond to the $M = 0.8, J = 10.2$ case. Right: Distributions for the $M = 0.8, J = 16.7$ case.

rest of the RANS parameters. The figure also shows the dramatic shrinkage of the (valid) parameter space / parametric uncertainty as a result of assimilation of experimental data.

We next examine the calibrated model's predictive skill. We use 100 $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ samples from the posterior distribution to generate vorticity fields on the crossplane by running the 3D RANS simulator. The vorticity field is then approximated by a point vortex, whose circulation is obtained by integrating the vorticity in $\mathcal{W}$. The centroid of the vorticity distribution in $\mathcal{W}$ is assigned the position (y, z) of the point vortex. In addition, we compute the radius of gyration of the vorticity distribution inside the $\mathcal{W}$ as a simple proxy for its size. We will refer to these metrics as the "point vortex metrics". In Fig. 7 (left), we boxplot the "point vortex metrics", normalized by their experimental counterparts for the $M = 0.8, J = 10.2$ case. The individual circles plot the predictions obtained using $\mathbf{C}_{nom}$. We see that for the circulation and the position, our errors are well within 5%; this holds true even for the interquartile range. The radius of gyration is underpredicted by about 10%. Such an accurate representation is somewhat expected since the vorticity field was used to calibrate the model. In Fig. 7 (right), we plot the results for the $M = 0.8, J = 16.7$ case. Note that the model was *not* calibrated for this case and the plot measures the predictive skill of the model at an off-calibration point. We see that the circulation and height y of the centroid are

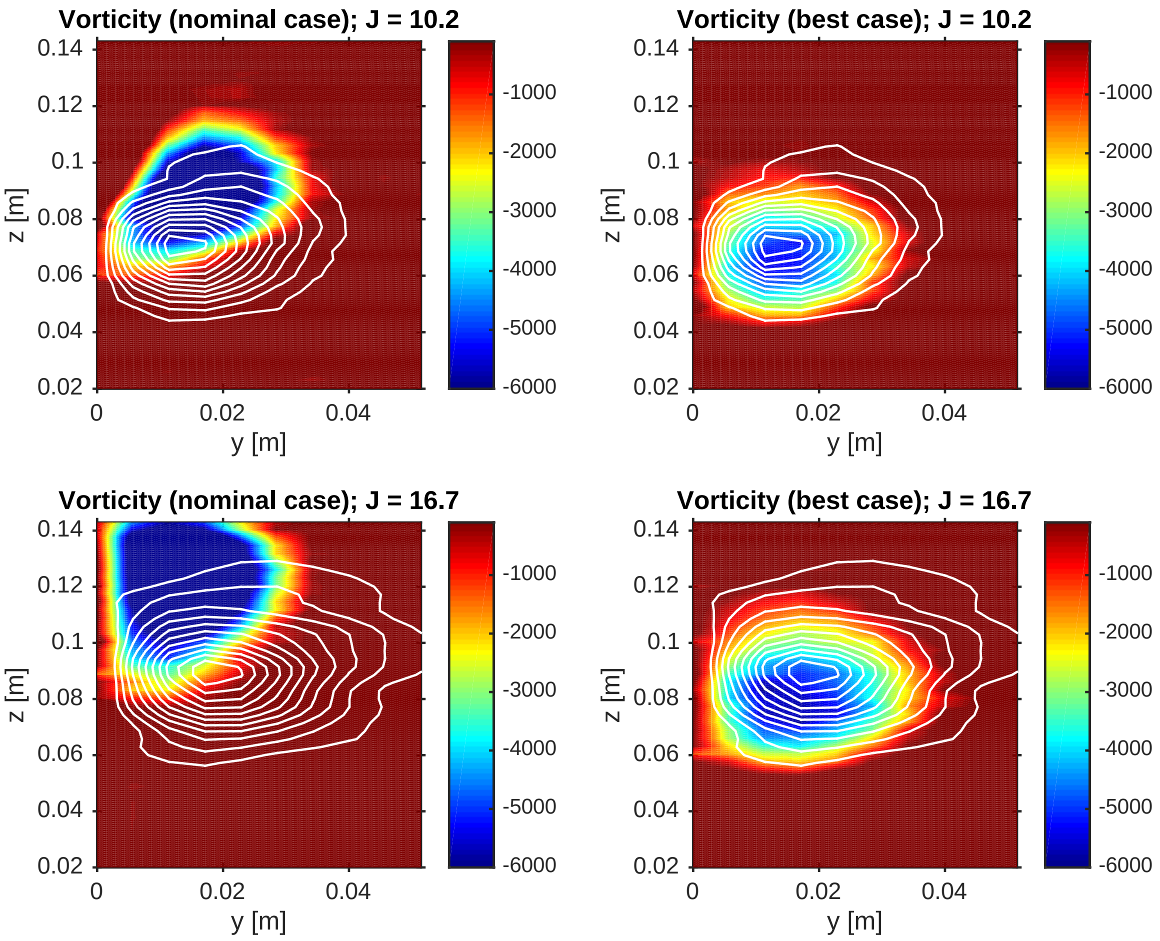

Fig. 8 Left column: Streamwise vorticity in the crossplane computed using $\mathbf{C}_{nom}$, overlaid with vorticity, as measured in experiments. $J = 10.2$ and 16.7 for the top and bottom rows. Right column: Crossplane streamwise vorticity predicted using the "optimal" values of $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1}), \mathbf{C}_{opt}$. Qualitatively, the agreement is quite good. The improvement over $\mathbf{C}_{nom}$ (left column) is stark. Linear dimensions are in meters.

accurately captured for the $J = 16.7$ case, but 20% - 30% errors are seen in z (horizontal distance from the test-section midplane) and the radius of gyration. The improvement of the predictions, post-calibration, are stark. The Beresh experiments do not contain crossplane measurements for the $M = 0.7, J = 10.2$ case and consequently, we omit the comparison here.

The set of runs plotted in Fig. 7, for the $M = 0.8, J = 10.2$ case, allow us to identify an optimal parameter combination (from the 100 points sampled from the posterior). The deviations (from 1) of the ratios plotted in Fig. 7 (left) allow us to define an error and consequently a mean square error that combines the deviations in circulation, centroidal position and the radius of gyration.

The run with the minimum mean squared error yields the best $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ combination $\mathbf{C}_{opt} = \{0.1025, 2.099, 1.416\}$. Note that this value is not very different from the optimal values yielded by the genetic algorithm optimization ($C_\mu = 0.105, C_{\epsilon 2} = 2.099, C_{\epsilon 1} = 1.42$). In Fig. 8, we plot, in the left column, the streamwise vorticity distribution computed on the right half of the crossplane using $\mathbf{C}_{nom}$. In the right column, we plot the corresponding vorticity distribution obtained using $\mathbf{C}_{opt}$. Overlaid on each are contour plots of the vorticity distribution obtained experimentally. The top row contains results for the $M = 0.8, J = 10.2$ case; $J = 16.7$ is plotted in the bottom row. In general, the vorticity distribution obtained using $\mathbf{C}_{opt}$ (right column) is very close to the experimental values. The improvement over the predictions with $\mathbf{C}_{nom}$ (left column) is substantial. These figures bear out the summaries in Fig. 7, which showed that calibration to the $M = 0.8, J = 10.2$ case improved the predictive skill of RANS at an off-calibration value of J.

In Fig. 9 we investigate the effect of the model-data mismatch σ^2. We use the 100 $\{C_\mu, C_{\epsilon 2}, C_{\epsilon 1}, \sigma^2\}$ samples from the posterior distribution to perform an ensemble of simulations and predict the crossplane vorticity at the 52 probes used in the calibration. These predictions, normalized by their experimental counterparts, are plotted in Fig. 9 (top left); the horizontal line indicates when the model predictions match experimental values. We call these the "pushed forward posterior". We also add the model-data mismatch (the effect of σ^2) to the vorticity predictions and plot them, normalized by the experimental values, in Fig. 9 (top right). These are conventional posterior predictive runs. The error bars are the 5th and 95th percentiles of the ensemble of model predictions. It is clear that without the effect of σ^2, there is hardly any variability in the vorticity predictions due to uncertainty in $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$; for probe numbers 1-30, the model ensemble predictions are also not close to the experimental values. It is only after adding the effect of the model-data mismatch σ^2 that the model predictions cover the experimental measurements as seen in Fig. 9 (top right). Thus, the data-model mismatch for vorticity, a composite of measurement, surrogate, discretization and model-form errors, is quite necessary for *covering* experimental data, not merely coming close to it. Further, since σ^2 is a measure of square-integrated differences between model predictions and experimental values, small spatial shifts/disagreements of the two rough fields can result in large σ^2. In contrast, for integrated (i.e., *not square-integrated*) quantities like circulation, the spatial shifts

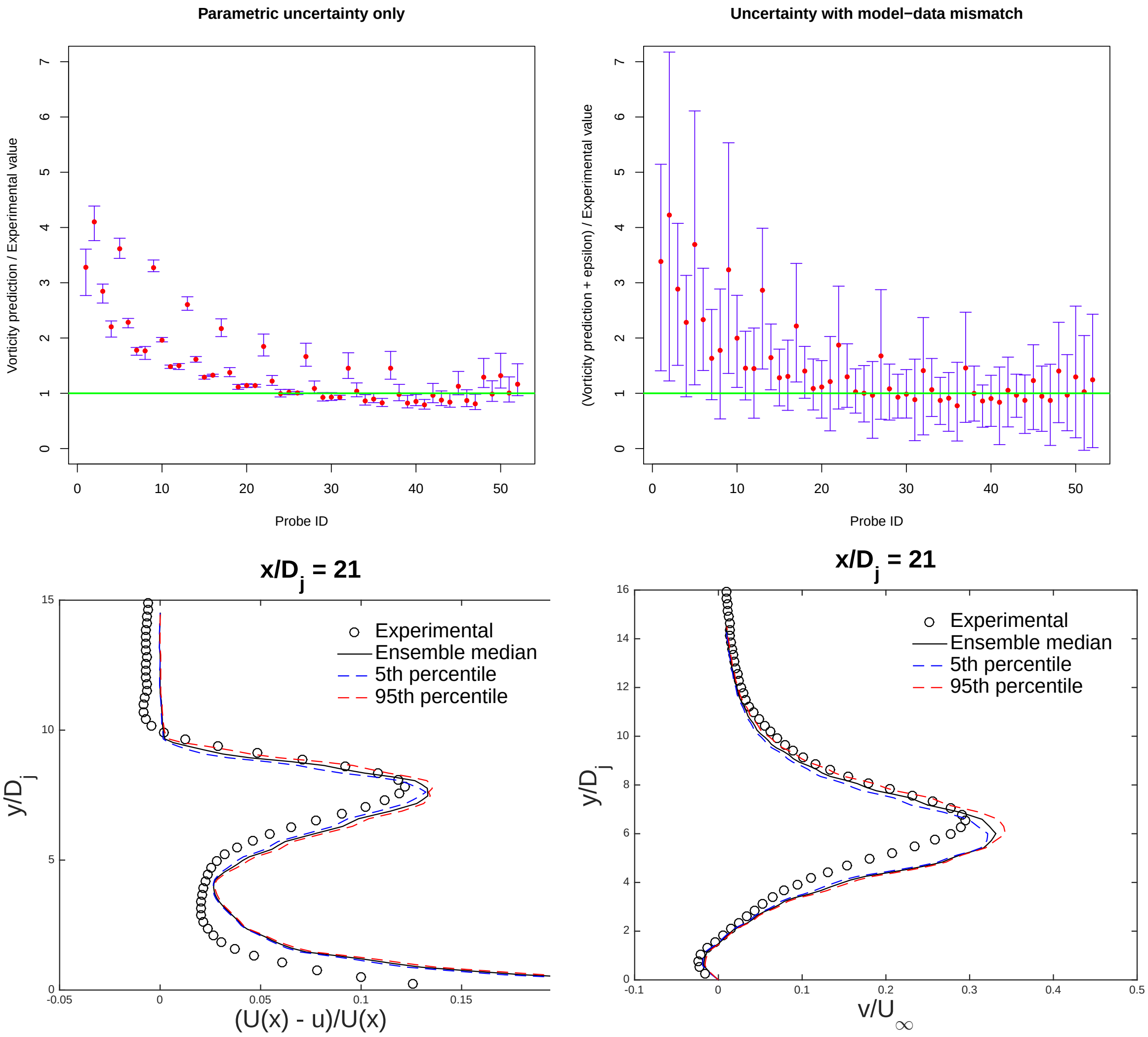

Fig. 9 Top row: Model predictions of crossplane vorticity at the 52 probes using 100 $\{C_\mu, C_{\epsilon 2}, C_{\epsilon 1}, \sigma^2\}$ samples drawn from the posterior distribution. Predictions are normalized by the corresponding experimental values. A value of 1 (horizontal line) indicates that predictions match experimental measurements. Error bars indicate the 5th and 95th percentile values computed using the 100 predictions. The filled symbols are the median predictions. On the left, we compute vorticity using $\sigma^2 = 0$ i.e., variability in model predictions is entirely due to uncertainty in $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ and have no contribution from the model-data mismatch. On the right, we add in the effect of σ^2. Bottom row: Predictions of the streamwise velocity deficit (left) and vertical velocity (right) on the midplane at a point 21 jet diameters downstream of the jet. Experimental measurements are plotted with symbols (○), and the median prediction from the 100 model simulations with solid lines (—) . The dashed lines (- - -) are the 5th and 95th percentiles of the model predictions. These predictions do not have any contributions of the model-data mismatch.

and rough nature cease to be significant due to cancellation of opposite-signed differences, and the agreement between numerical predictions and experimental values is quite good (see Fig. 7).

In Fig. 9 (bottom row) we plot summaries of the ensemble of model predictions of streamwise velocity deficit and vertical velocities on the midplane at $x/D_j = 21$, where D_j is the jet diameter. The median predictions are plotted with solid lines and the 5th and 95th percentiles with dashed lines. Since these measurements were not used in the calibration, we do not have a model-data mismatch for them and these simulations are "pushed forward posteriors". The qualitative features of the vorticity "pushed forward posteriors" are even more evident here - parametric uncertainty around the calibrated parameter values result in little predictive variability and do not cover the experimental measurements. However, calibration does draw the predictions closer to experimental measurements versus $\mathbf{C}_{nom}$; see Fig. 7, Fig. 8, Fig. 10 and Fig. 11.

In Fig. 10 (top), we plot the streamwise velocity deficit in the midplane at $x/D_j = 21$ for $M = 0.8, J = 10.2$, $M = 0.8, J = 16.7$ and $M = 0.7, J = 10.2$ interactions. In Fig. 10 (bottom) we plot the vertical velocity on the midplane at the same flow conditions. In Fig. 11, we plot the same quantities but at $x/D_j = 31.5$. The data for the plots were obtained using $\mathbf{C}_{nom}$, $\mathbf{C}_{opt}$ and the ensemble mean of the 100 $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ samples drawn from the posterior. These are compared with the experimental values ($\circ$). We see that the ensemble mean and $\mathbf{C}_{opt}$ predictions always lie closer to the experimental values than $\mathbf{C}_{nom}$; in the case of $M = 0.8, J = 10.2$ and $M = 0.7, J = 10.2$, the agreement between the predicted and experimental values is very close. Further, the predictions using $\mathbf{C}_{opt}$ and the ensemble mean are almost identical. Predictions using $\mathbf{C}_{nom}$ bear out the inferences drawn from Fig. 7 - the CVP sits far above the experimentally observed CVP, and is much too strong in all cases. Post-calibration, the position as well as the strength of the CVP is closer to the experimental values. The vertical velocity in the midplane is largely governed by the CVP. Fig. 7 and Fig. 8 show that the calibration immensely improves our ability to capture the crossplane vorticity (vis-à-vis predictions with $\mathbf{C}_{nom}$), and improvements in predictions of vertical velocity largely follow. Note that the calibration was performed only on the $M = 0.8, J = 10.2$ case, and that too, using streamwise vorticity on the crossplane; however, the calibration improves the entire flowfield (as seen in the streamwise deficit and vertical velocity profiles) and is applicable at

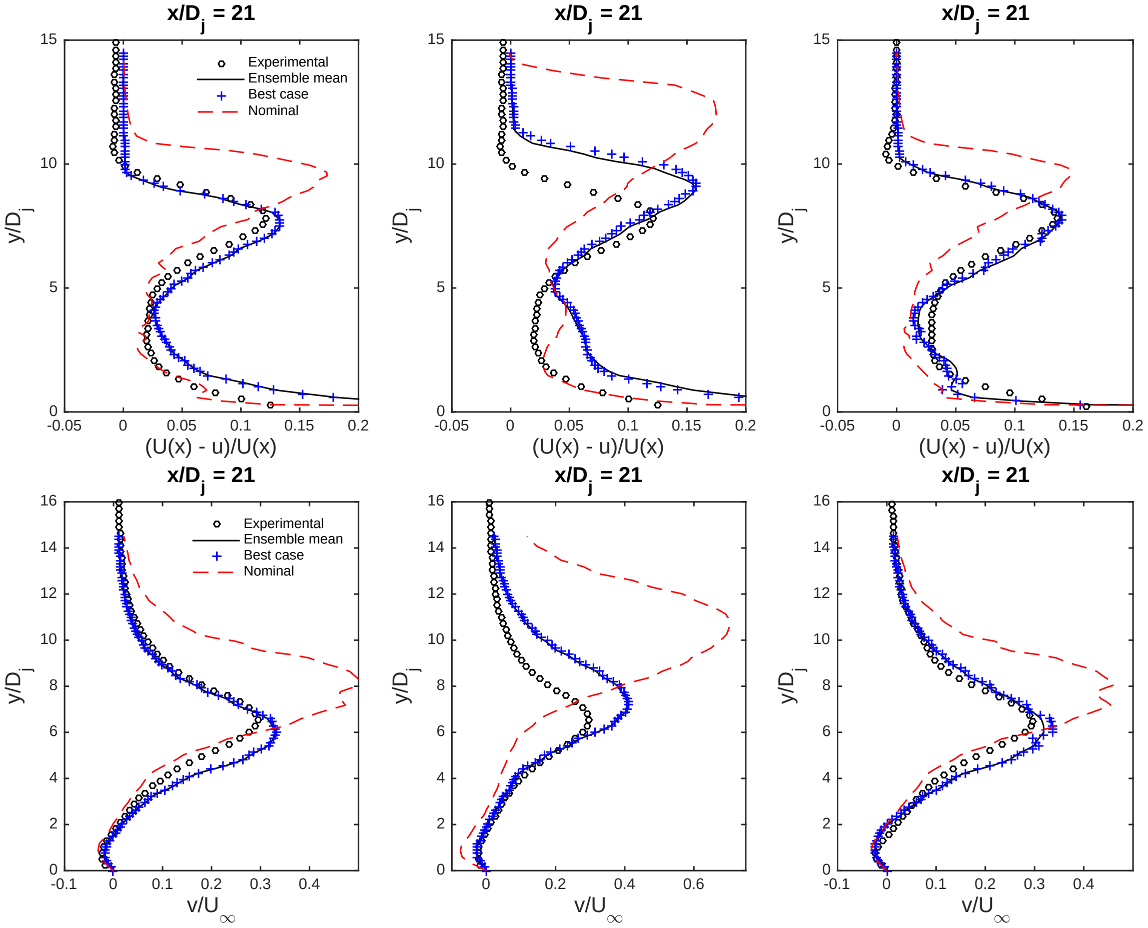

Fig. 10 Top row: Streamwise velocity deficit at $x/D_j = 21$, as computed for the $M = 0.8, J = 10.2$ (left column), $M = 0.8, J = 16.7$ (middle column) and $M = 0.7, J = 10.2$ (right column). In each subfigure on the top row, we plot the streamwise velocity deficit $(U(x) - u)/U(x)$ as measured experimentally (○), as predicted using $\mathbf{C}_{nom}$ (dashed red line, - - -) and $\mathbf{C}_{opt}$ (+). Here $U(x)$ is the bulk (or plug) flow velocity. We also plot the ensemble mean predictions from the 100 $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ samples from the posterior distribution as the thin black line — (it is very close to the $\mathbf{C}_{opt}$ prediction). Bottom row: We compare the vertical velocity v/U_∞ instead of streamwise velocity deficit.

other values of J and M as well.

Post-calibration, the main cause of the discrepancy between experimental observations and numerical predictions is the model-form error in RANS; the surrogate modeling error is a secondary source of error as it is capped at 10%. These errors are caused by the manner in which the turbulent stresses and their evolution are modeled. We use a linear eddy viscosity model, which relates the

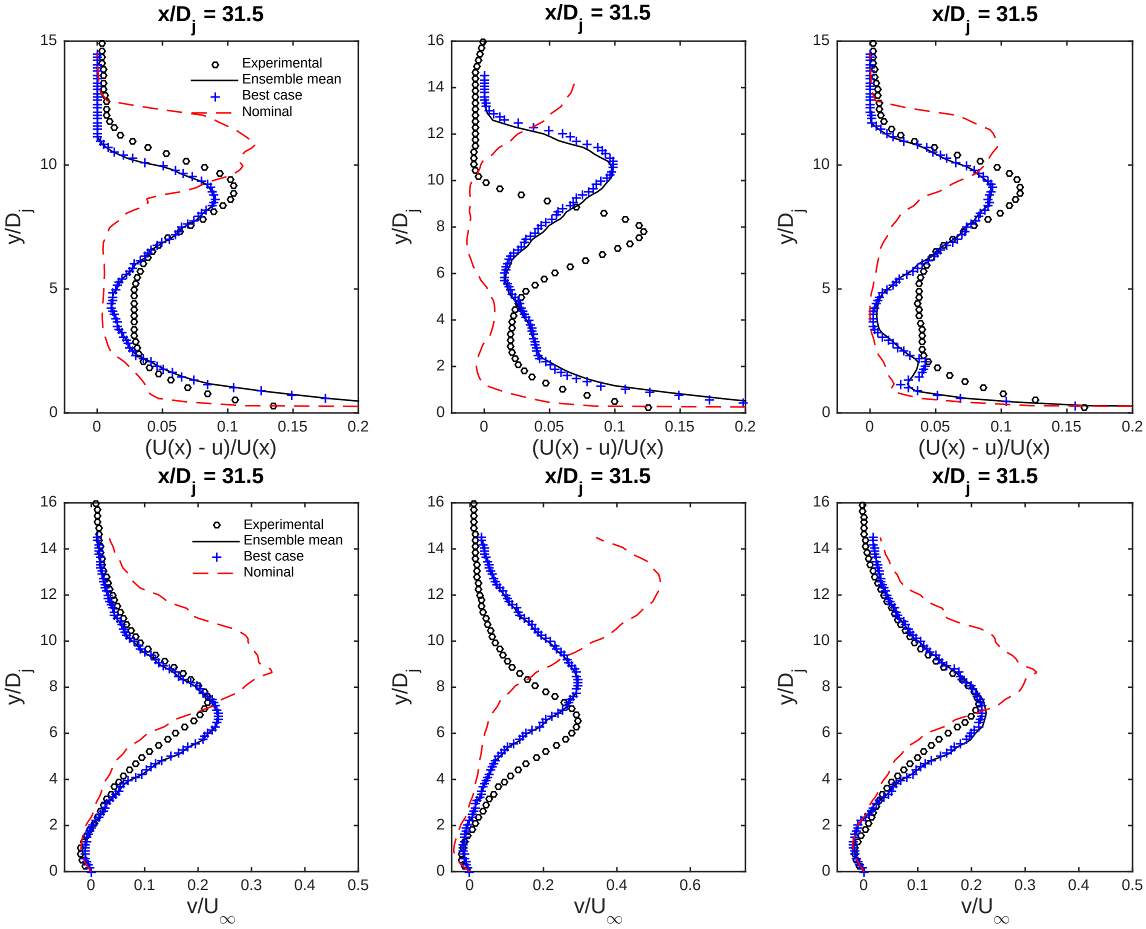

Fig. 11 Top row: Streamwise velocity deficit at $x/D_j = 31.5$, as computed for the $M = 0.8, J = 10.2$ (left column), $M = 0.8, J = 16.7$ (middle column) and $M = 0.7, J = 10.2$ (right column). In each subfigure on the top row, we plot the streamwise velocity deficit $(U(x) - u)/U(x)$ as measured experimentally ($\circ$), as predicted using $\mathbf{C}_{nom}$ (dashed red line, - - -) and $\mathbf{C}_{opt}$ ($+$). Here $U(x)$ is the bulk (or plug) flow velocity. We also plot the ensemble mean predictions from the 100 $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ samples from the posterior distribution as the thin black line — (it is very close to the $\mathbf{C}_{opt}$ prediction). Note that for the $M = 0.8, J = 16.7$ case, the nominal prediction of the streamwise flow deficit lies mostly outside the vertical range where experimental data was gathered. Bottom row: We compare the vertical velocity v/U_∞ instead of streamwise velocity deficit.

turbulent stresses τ_{ij} to strain-rates S_{ij} and turbulent kinetic energy k. It is stated as

$$\tau_{ij} = \overline{-u'_i u'_j} = \frac{2}{3} k \delta_{ij} - 2\nu_T S_{ij}, \qquad k = \sum_i \overline{u_i^2} \qquad \text{and} \qquad S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{1}{3}\delta_{ij}\sum_k \frac{\partial u_k}{\partial x_k} \quad (7)$$

and forms a convenient place to quantify and analyze the model-form error of our RANS model after calibration. In Fig. 12, we compare ensemble mean predictions of turbulent stresses obtained from the 100 $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ posterior samples against experimental measurements from Ref. [18] for our two tests. We also plot the predictions obtained using $\mathbf{C}_{nom}$. Calibration to crossplane vorticity (for the $M = 0.8, J = 10.2$ case) results in a loss of agreement with measurements for τ_{22} (under-prediction) and τ_{11} (over-prediction). τ_{12} is grossly underpredicted. In fact, the ensemble mean predictions of τ_{11} and τ_{22} are almost indistinguishable. The predictions with $\mathbf{C}_{nom}$ show the CVP sitting too high. The same behavior is seen at the off-calibration point - there is agreement for τ_{22} but the rest of the turbulent stresses are badly predicted. This behavior pinpoints the source of the model-form error.

The reason the ensemble mean predictions of τ_{ii} are so close is because the strain rates (in the midplane) in the RANS predictions are too small. This leads to a gross under-prediction of τ_{12} using Eq. 7 (see Fig. 12) whereas as for τ_{ii}, k dominates over the term with S_{ij}. Calibration leads to a k that splits the difference between τ_{11} and τ_{22}. The linear eddy viscosity model has too simple a form to capture the significant differences in the magnitude of the three stresses and a more complex model is required to bridge this gap. Yet, despite these large errors in the turbulent stress predictions, the large scale structures in the mean flow were only moderately affected, suggesting that the effect of these stresses, and the model-form error in the linear eddy viscosity model, are somewhat muted. Note that it is the gradients of the turbulent stresses, and not the stresses themselves, that appear in conservation equations for momentum and energy.

V. Conclusions

In this study, we have investigated whether RANS models can be calibrated to be predictive in supersonic-jet-in-transonic-crossflow interactions. We have proceeded under the hypothesis that the nominal values of $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1}) = \mathbf{C}_{nom}$ are unsuitable and a better parameter set could be identified via calibration to experimental data. We find that it is possible to do so; further, the calibrated RANS model is predictive for JIC interactions different from the one employed in the calibration.

We adopted a Bayesian calibration method and estimated the RANS parameters $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$ as a joint PDF, thus quantifying the uncertainty in the estimation. We employed an adaptive

Markov chain Monte Carlo (MCMC) method to do so. We also obtained a crude estimate of the data - model mismatch, an amalgamation of measurement, surrogate modeling and model-form errors; it was represented statistically as a Gaussian distribution with zero mean and a standard deviation of σ^2. Vorticity on the crossplane was used as the observable, and was found to be sufficient for estimating $(C_\mu, C_{\epsilon2}, C_{\epsilon1})$. The calibration used surrogates of the 3D RANS simulator, due to the phenomenal computational cost of using MCMC. Construction of the surrogates posed a formidable challenge. While the bounds on $C_\mu, C_{\epsilon2}$ and $C_{\epsilon1}$ are known, their random combinations do not yield physically realistic flows, and in some cases, the RANS simulator will not even yield results. We identified a physically realistic region $\mathcal{R}$ of the parameter space and developed polynomial surrogates that were valid only inside $\mathcal{R}$. The region $\mathcal{R}$ is complex and we developed a support vector machine classifier to identify whether an arbitrary $(C_\mu, C_{\epsilon2}, C_{\epsilon1})$ combination was inside it. $\mathcal{R}$ was also leveraged to serve as an informative, if improper, prior for $(C_\mu, C_{\epsilon2}, C_{\epsilon1})$; prior density was 1 inside $\mathcal{R}$ and zero outside. The classifier, as well as the polynomial surrogates, were used with MCMC to obtain a 4-dimensional joint PDF for $(C_\mu, C_{\epsilon2}, C_{\epsilon1}, \sigma^2)$.

The joint PDF for $(C_\mu, C_{\epsilon2}, C_{\epsilon1}, \sigma^2)$ was constructed using measurements from an experiment conducted at $M = 0.8, J = 10.2$ [17–19]. Bayesian calibration dramatically shrank the valid region of the parameter space, reducing parametric and predictive uncertainties. MAP estimates of $C_{\epsilon2}$ and $C_{\epsilon1}$ are similar to those obtained in Ref. [42]. However, the MAP estimate of C_μ and the PDFs do not agree with those in Ref. [42]; given the difference in the flow regimes and geometries, this is not very surprising. As a test of the predictive skill of the posterior PDF, 100 samples drawn from it were used to run the 3D RANS simulator and predict the vorticity on the crossplane. Prediction errors were $\leq 5\%$ for integrated quantities such as circulation, which was expected since vorticity was the calibration variable. However, the same runs yielded very good agreement with the experimentally measured streamwise velocity deficit and vertical velocity on the midplane, providing some corroboration that the flowfield away from the crossplane was similar to the experimental one. The improvement over the flowfield computed with the nominal values of $(C_\mu, C_{\epsilon2}, C_{\epsilon1})$, $\mathbf{C}_{nom}$ is substantial, and can be seen in all the flow variables that we investigated. We used the joint PDF, computed using $M = 0.8, J = 10.2$ data, to predict the flowfield at $M = 0.8, J = 16.7$ and

$M = 0.7, J = 10.2$. Our calibrated RANS model yielded substantial improvements over predictions using $\mathbf{C}_{nom}$. Thus our hypothesis that the inaccuracy of RANS in jet-in-crossflow interactions was due to the unsuitability of $\mathbf{C}_{nom}$ proved correct; in comparison, the model-form error was of secondary importance. Note that while we may have obtained encouraging results for JIC, we do not claim that our PDF will be equally predictive for other flow configurations.

The model-form error, nevertheless, does exist, and post-calibration, is the dominant error term. It causes the degradation of predictive skill when we apply our RANS model at off-calibration points. We isolated the effect of the model-form error on the midplane and pinpointed its source to be the simplicity of the linear eddy viscosity model used in our simulations. The contributions to the turbulent stresses by mean flow quantities (strain rate in the case of the linear eddy viscosity model) are grossly underpredicted. This error could be rectified by augmenting this closure model, for example, by employing a nonlinear eddy viscosity model. This investigation is left for future work.

Acknowledgment

This work was supported by Sandia National Laboratories' Advanced Scientific Computing (ASC) Verification and Validation program. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U. S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. We thank Lawrence Livermore National Laboratory, Livermore, CA, for computer time on the Sequoia supercomputer, a National User Facility.

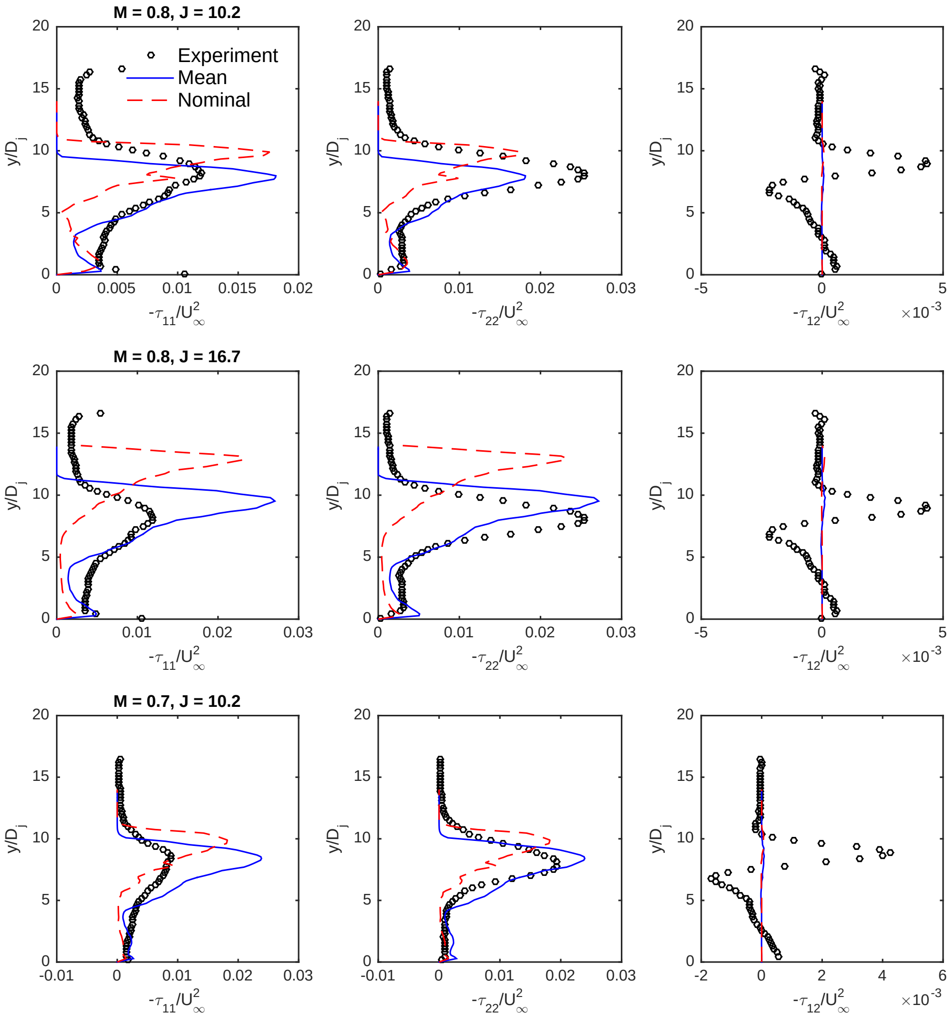

Fig. 12 Profiles of $-\tau_{11}/U_\infty^2$ (first column), $-\tau_{22}/U_\infty^2$ (second column) and $-\tau_{12}/U_\infty^2$ (last column) as a function of y/D_j at $x = 200$ mm behind the jet on the mid-plane ($x/D_j = 20.9$). Experimental measurements are plotted with a ∘, the ensemble mean predictions from our 100 $((C_\mu, C_{\epsilon 2}, C_{\epsilon 1}))$ values from the posterior are plotted with a solid line (–) and the predictions with C$_{nom}$ are plotted with a dashed line (- - -). Top row, we plot results from the calibration case ($M = 0.8, J = 10.2$). The middle row contains results from the ($M = 0.8, J = 16.7$) test case. The bottom row contains results from the $M = 0.7, J = 10.2$ case.

References

[1] Wilcox, D. C., *Turbulence Modeling for CFD*, D C W Industries, 1998.

[2] Poroseva, S. and Iaccarino, G., "Simulating separated flows using the $k-\epsilon$ model," Tech. rep., Center for Turbulence Research, Annual Research Briefs, 2001.

[3] Kimura, I. and Hosoda, T., "A non-linear $k-\epsilon$ model with realizability for prediction of flows around bluff bodies," *International Journal for Numerical Methods in Fluids*, Vol. 42, 2003, pp. 813–837.

[4] Sjögren, T. and Johansson, A. V., "Development and calibration of algebraic nonlinear models for terms in the Reynolds stress transport equation," *Physics of Fluids*, Vol. 12, No. 6, 2000, pp. 1554–1572.

[5] Durbin, P. A., "Separated flow computations using the $k-\epsilon-v^2$ model," *AIAA Journal*, Vol. 33, 1995, pp. 659–664.

[6] Yakhot, V. and Orszag, S., "Renormalization group analysis of turbulence," *Physical Review Letters*, Vol. 57, 1986, pp. 1722–1724.

[7] Shih, T. H., Liou, W. W., Shabbir, A., and Yang, Z., "A new $k-\epsilon$ eddy viscosity model for high Reynolds number turbulence flows," *Computers and Fluids*, Vol. 24, 1995, pp. 227–238.

[8] Lumley, J. L., "Computational modeling of turbulent flows," *Advances in Applied Mechanics*, Vol. 18, 1978, pp. 123–177.

[9] Grimaji, S. S., "Pressure-strain correlation modelling of complex turbulent flows," *Journal of Fluid Mechanics*, Vol. 422, 2000, pp. 91–123.

[10] Arunajatesan, S., "Evaluation of two-equation RANS models for simulation of jet-in-crossflow problems," *50th AIAA Aerospace Sciences Meeting*, 2012.

[11] Ray, J., Lefantzi, S., Arunajatesan, S., and Dechant, L., "Bayesian calibration of a $k-\epsilon$ model for predictive jet-in-crossflow simulations," *Proceedings of the 44th AIAA Fluid Dynamics Conference*, 2014, Paper No. AIAA-2014-2085.

[12] Viana, F. A. C., Simpson, T. W., Balabanov, V., and Toporov, V., "Metamodeling in multidisciplinary design optimization: How far have we really come?" *AIAA Journal*, Vol. 52, 2014, pp. 670–690.

[13] Sargsyan, K., Safta, C., Najm, H. N., Debusschere, B. J., Ricciuto, D., and Thornton, P., "Dimensionality reduction for complex models via Bayesian compressive sensing," *International Journal for Uncertainty Quantification*, Vol. 4, No. 1, 2014, pp. 63–93.

[14] Giunta, A. A., Narducci, R., Burgee, S., Grossman, B., Haftka, R. T., Mason, W. H., and Watson, L. T., "Variable-complexity response surface aerodynamic design of an HSCT wing," *Proceedings of the 13th Applied Aerodynamics Conference*, 1995, Paper No. AIAA-1995-1886.

[15] Mahesh, K., "The interaction of jets with crossflows," *Annual Reviews of Fluid Mechanics*, Vol. 45, 2013, pp. 379–407.

[16] Peterson, C. W., Wolfe, W. P., and Payne, J. L., "Experiments and computations of roll toque inducd by vortex-fin interaction," *42nd Aerospace Sciences Meeting and Exhibit, Reno, Nevada*, 2004.

[17] Beresh, S. J., Henfling, J. F., Erven, R. J., and Spillers, R. W., "Penetration of a transverse supersonic jet into a subsonic compressible crossflow," *AIAA Journal*, Vol. 43, No. 2, 2005, pp. 379–389.

[18] Beresh, S. J., Henfling, J. F., Erven, R. J., and Spillers, R. W., "Turbulent characteristics of a transverse supersonic jet in a subsonic compressible crossflow," *AIAA Journal*, Vol. 43, No. 11, 2005, pp. 2385–2394.

[19] Beresh, S. J., Henfling, J. F., Erven, R. J., and Spillers, R. W., "Crossplane velocimetry of a transverse supersonic jet in a transonic crossflow," *AIAA Journal*, Vol. 44, No. 12, 2006, pp. 3051–3061.

[20] Beresh, S. J., Heineck, J. T., Walker, S. M., Schairer, E. T., and Yaste, D. M., "Planar velocimetry of jet/fin interaction on a full-scale flight vehicle configuration," *AIAA Journal*, Vol. 45, No. 8, 2007, pp. 1827–1840.

[21] Chai, X. and Mahesh, K., "Simulations of high speed turbulent jets in crossflow," *49th AIAA Aerospace Sciences Meeting, Orlando Florida*, 2011.

[22] Genin, F. and Menon, S., "Dynamics of sonic jet injection into supersonic crossflow," *Journal of Turbulence*, Vol. 11, 2010, pp. 1–13.

[23] Rodebaugh, G. P., Brinckman, K. W., and Dash, S. M., "DDES of aeropropulsive flows based on an extended $k-\epsilon$ RANS model," *51st AIAA Aerospace Sciences Meeting, Grapevine, Texas*, 2013.

[24] Brinckman, K. W., Calhoon, W. H., and Dash, S. M., "Scalar fluctuation modeling for high-speed aeropropulsive flows," *AIAA Journal*, Vol. 45, No. 5, 2007, pp. 1036–1046.

[25] Papp, J. L. and Dash, S. M., "Turbulence model unification and assessment for high-speed aeropropulsive flows," *Proceedings of the 39th AIAA Aerospace Sciences Meeting and Exhibit*, 2001, Paper No. AIAA-2001-0880.

[26] Edeling, W. N., Cinnella, P., Dwight, R. P., and Bijl, H., "Bayesian estimates of parameter variability in $k-\epsilon$ turbulence model," *Journal of Computational Physics*, Vol. 258, 2014, pp. 73–94.

[27] Mohamed, M. S. and Larue, J. C., "The decay power law in grid-generated turbulence," *Journal of Fluid Mechanics*, Vol. 219, No. 1, 1990, pp. 195–214.

[28] Kim, J., Moin, P., and Moser, R., "Turbulence statistics in fully developed channel flow at low Reynolds number," *Journal of Fluid Mechanics*, Vol. 177, No. 1, 1987, pp. 133–166.

[29] Tavoularis, S. and Karnik, U., "Further experiments on the evolution of turbulent stresses and scales in uniformly sheared flows," *Journal of Fluid Mechanics*, Vol. 204, No. 1, 1989, pp. 457–478.

[30] Platteeuw, P. D. A., Loeven, G. J. A., and Bijl, H., "Uncertainty quantification applied to the $k - \epsilon$ model of turbulence using the probabilistic collocation method," *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 2008.

[31] Emory, M., Pecnik, R., and Iaccarino, G., "Modeling structural uncertainties in Reynolds-Averaged computations of shock/boundary layer interactions," *49th AIAA Aerospace Sciences Meeting*, 2011.

[32] Gorlé, C. and Iaccarino, G., "A framework for epistemic uncertainty quantification of turbulent scalar flux models for Reynolds-averaged Navier-Stokes simulations," *Physics of Fluids*, Vol. 25, 2013, pp. 055105.

[33] Oliver, T. A. and Moser, R. D., "Bayesian uncertainty quantification applied to RANS turbulence models," *Journal of Physics: Conference Series*, Vol. 318, 2011, pp. 042032.

[34] Cheung, S. H., Oliver, T. A., E. E. Prudencio, S. P., and Moser, R. D., "Bayesian uncertainty analysis with applications to turbulence modeling," *Reliability Engineering and System Safety*, Vol. 96, 2011, pp. 1137–1149.

[35] Dow, E. and Wang, Q., "Quantification of structural uncertainties in the $k - \omega$ turbulence model," *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 2011.

[36] Edeling, W. N., Cinnella, P., and Dwight, R. P., "Predictive RANS simulations via Bayesian Model-Scenario Averaging," *Journal of Computational Physics*, Vol. 275, 2014, pp. 65–91.

[37] Duraisamy, K., Zhang, Z. J., and Singh, A. P., "New approaches in turbulence and transition modeling using data-driven techniques," *Proceedings of the 53rd AIAA Aerospace Sciences Meeting, Kissimmee, FL*, 2015, Paper No. AIAA-2015-1284.

[38] Tracey, B., Duraisamy, K., and Alonso, J. J., "A machine learning strategy to assist turbulence model development," *Proceedings of the 53rd AIAA Aerospace Sciences Meeting, Kissimmee, FL*, 2015, Paper No. AIAA-2015-1287.

[39] Zhang, Z. J. and Duraisamy, K., "Machine learning methods for data-driven turbulence modeling," *Proceedings of the 22nd AIAA Computational Fluid Dynamics Conference*, 2015, Paper No. AIAA-2015-2460.

[40] Parish, E. and Duraisamy, K., "Quantification of turbulence modeling uncertainties using full field inversions," *Proceedings of the 22nd AIAA Computational Fluid Dynamics Conference*, 2015, Paper No. AIAA-2015-2459.

[41] Jouhaud, J.-C., Sagaut, P., Enaux, B., and Laurenceau, J., "Sensitivity analysis and multiobjective optimization for LES numerical parameters," *Journal of Fluid Engineering*, Vol. 130, 2008, pp. 021401.

[42] Guillas, S., Glover, N., and Malki-Epshtein, L., "Bayesian calibration of the constants of the $k - \epsilon$ turbulence model for a CFD model of street canyon flow," *Computer Methods in Applied Mechanics and Engineering*, Vol. 279, 2014, pp. 536–553.

[43] Hastie, T., Tibshirani, R., and Friedman, J., *The elements of statistical learning*, Springer Series in Statistics, Springer, 2nd ed.

[44] Gilks, W. R., Richardson, S., and Spiegelhalter, D. J., editors, *Markov Chain Monte Carlo in Practice*, Chapman & Hall, 1996.

[45] Silverman, B. W., *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, 1986.

[46] Haario, H., Laine, M., Mira, A., and Saksman, E., "DRAM-Efficient adaptive MCMC," *Statistics and Computing*, Vol. 16, No. 4, 2006, pp. 339–354.

[47] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2012, ISBN 3-900051-07-0.

[48] Soetaert, K. and Petzoldt, T., "Inverse Modelling, Sensitivity and Monte Carlo Analysis in R Using Package FME," *Journal of Statistical Software*, Vol. 33, No. 3, 2010, pp. 1–28.

[49] Raftery, A. and Lewis, S. M., "Implementing MCMC," *Markov Chain Monte Carlo in Practice*, edited by W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, Chapman and Hall, 1996, pp. 115–130.

[50] Warnes, G. R. and with contributions by Robert Burrows, *mcgibbsit: Warnes and Raftery's MCGibbsit MCMC diagnostic*, 2011, R package version 1.0.8.

[51] Dutang, C. and Savicky, P., *randtoolbox: Generating and Testing Random Numbers*, 2013, R package version 1.13.

[52] Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., and Leisch, F., *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*, 2014, R package version 1.6-3.

[53] Scrucca, L., "GA: A Package for Genetic Algorithms in R," *Journal of Statistical Software*, Vol. 53, No. 4, 2013, pp. 1–37.

[54] Yu, X. and Gen, M., *Introduction to evolutionary computing*, Springer-Verlag, Berlin, 2010.

[55] Cowles, M. K. and Carlin, B. P., "Markov chain Monte Carlo convergence diagnostics: A comparative review," *Journal of the American Statistical Association*, Vol. 91, No. 434, pp. 883–904.

[56] Ray, J., Pinar, A., and Seshadhri, C., "A stopping criterion for Markov chains when generating independent random graphs," *Journal of Complex Networks*, Vol. 3, No. 2, 2015, pp. 204–220.

Appendix A: Stopping criterion for the MCMC procedure

In this section, we analyze the chains from the MCMC inference to gauge their convergence to a 4-dimensional stationary distribution in the $\{C_\mu, C_{\epsilon 2}, C_{\epsilon 1}, \sigma^2\}$ space. The MCMC chain was run to 40,000 iterations, regardless of whether $\mathbf{y}_e$ consisted of synthetic or experimental observations. In this analysis we use the case where $\mathbf{y}_e$ consisted of measurements from the Beresh experiments, since it provides a more stringent convergence test case. 1 in every 5 samples was retained i.e., the MCMC chain was thinned by a factor of 5. The thinned chain of 8000 samples is saved to a file. In Fig. 13 (top row) we plot the last 400 samples of the thinned MCMC chains for $(C_\mu, C_{\epsilon 2}, C_{\epsilon 1})$; the auto-correlation is quite evident. In the bottom row, we plot the partial auto-correlation functions for the same chains. We see the dominant auto-correlation is of lag 1 (or a lag of 5 in the un-thinned Markov chains). However, some auto-correlation persists till about a lag of 3 or 4 (15-20 in the un-thinned chains).

The convergence of the chains was tested using the Raftery-Lewis (RL) method [49], as implemented in `mcgibbsit` package [50] in `R`. Convergence of the median, the most stringent criterion for the RL test [55], was used as the convergence metric. The RL test uses the median of each parameter to convert the corresponding chain to a binary (0/1) one. It then counts the 0-0, 0-1, 1-1 and 1-0 transitions to fill a contingency table. The table is checked to see if the transitions resemble draws from a high-order Markov process (i.e., auto-correlated draws) versus a first-order one; if so, the chain is thinned, usually by a factor of 2, and the test is repeated. The testing process stops when the transitions in the thinned chain resemble a first-order Markov process more than competing models. At that point, the test checks if the thinned chain has sufficient samples to compute the median with the desired accuracy which is specified during the test; we use a band of ± 0.025 around the median ($q = 0.5$) as the test of estimation accuracy. If the chain fails the test, the RL method calculates the required number of samples, and consequently the desired length of the MCMC chain. A more complete description is in Ref. [56] and references therein.

The RL test applied to the thinned-by-a-factor-of-5 chains identifies that the most auto-correlated chain (that of $C_{\epsilon 1}$) has an autocorrelation of 3.68. It also identifies that 5652 out of the 8000 samples or a chain of $5652 \times 5 = 28,260$ MCMC steps would be sufficient to estimate

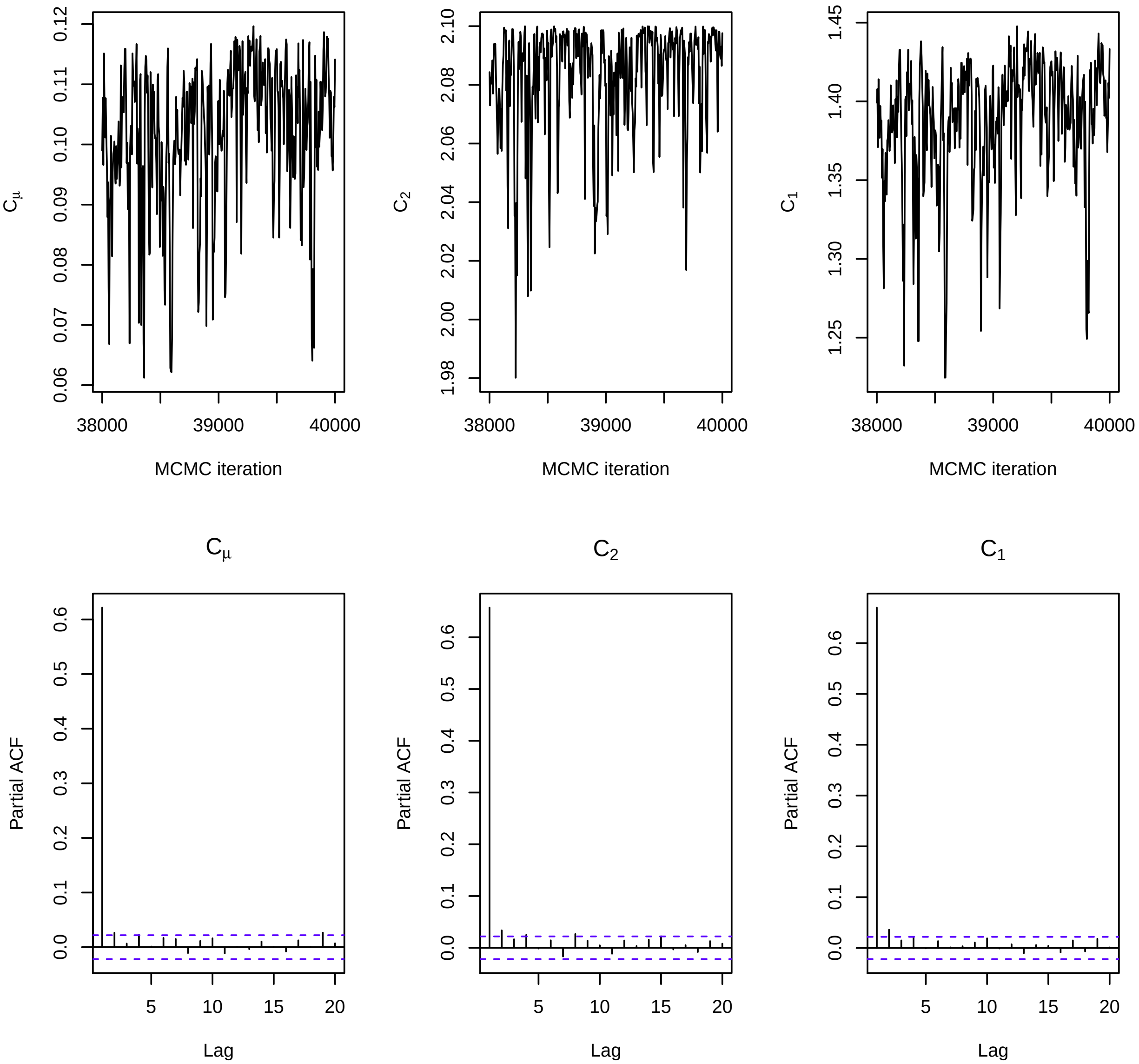

Fig. 13 Top row: Plots of the thinned MCMC chains for $C_\mu, C_{\epsilon 2}$ and $C_{\epsilon 1}$. The chains show significant auto-correlation (PACF). Bottom row: Partial auto-correlation functions for the three chains. We see that auto-correlation persists till about a lag of 3. The dashed line indicates a statistically significant PACF level

the median with the requisite accuracy. This number (i.e., 28,000) is reported in Sec. IV as the number of MCMC steps required for convergence. The convergence efficiency could be improved by re-parameterizing the problem; however, each run with surrogate models takes about 2-3 hours and reducing the run-time was not a priority.